Oracle Discoverer for REMIS

QualityNet Conference
September 9 – 13, 2002
Hunt Valley, Maryland

Oracle Discoverer for REMIS
Course Content

Oracle Discoverer is an ad hoc query tool that enables end users to selectively retrieve data from relational databases. The REMIS application already presents data to you in a logical manner, however, for reporting or further analysis, ad hoc query tools become a necessity.

If you are familiar with other ad hoc query tools such as InfoMaker, Microsoft Access, or SAS, you know that a level of understanding of relational databases is necessary to code the proper SQL syntax, join multiple tables together, and select columns from tables, all with very cryptic names.

With Oracle Discoverer, you do not have to be a programmer or DBA in order to create queries. You should find your data grouped together meaningfully, with names that don't look like they came from a Programming 101 course manual. The final hurdle that may need to be overcome is how to actually use those groups of data, and select only that information you want to see. That is the goal of this class – to help you convert your statement of business need into a Discoverer query with conditions set to meet your needs. Our goal is to give you a basic foundation of this tool so that your comfort level will grow in the weeks to come.

Topic 1 - General Overview

A few pre-defined queries have been created for you in Discoverer. These queries were modeled after reports available to you in REBUS and consist of the following:

- Patient Summary Report
- Beneficiary Entitlement Summary
- Transplant Report

Discoverer queries are stored in worksheets that are grouped together in workbooks, just as you would find in Microsoft Excel. Each workbook may contain more than one worksheet, and each worksheet usually contains an entirely different query than its companion worksheets in the workbook. If a workbook has more than one worksheet, a series of tabs across the bottom of the window name the various worksheets.

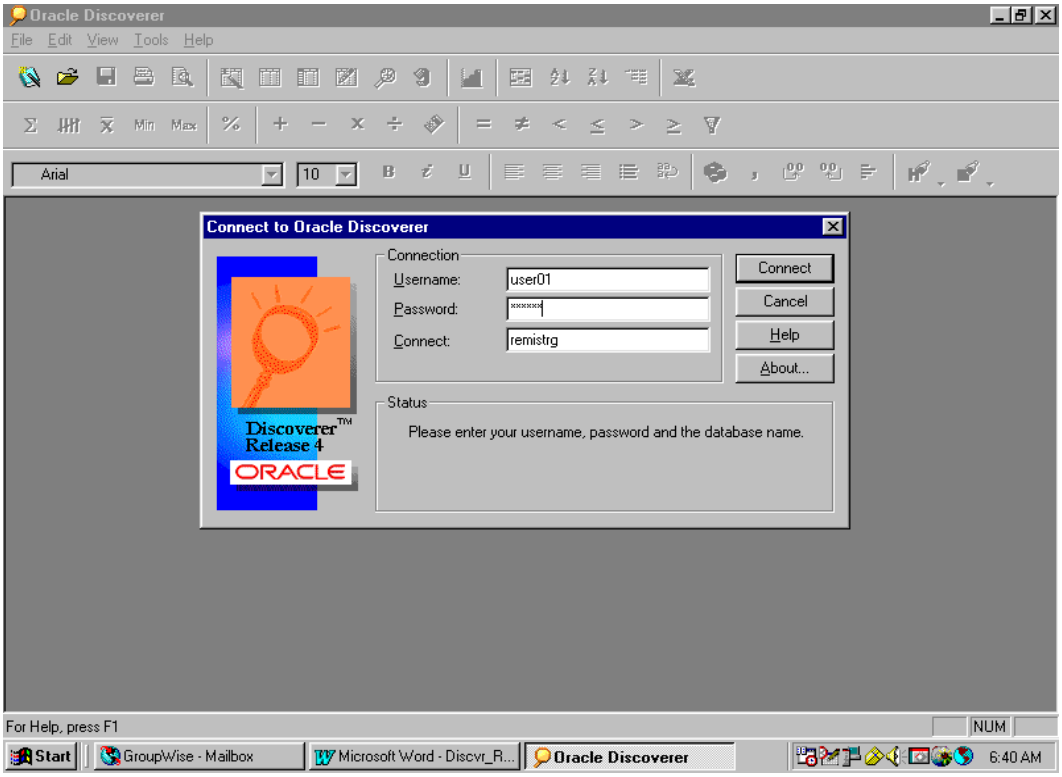
When you open and run one of the predefined workbooks, you will not be working with the original workbook. Instead, you are working with a copy of that workbook. You may save your copy when you are finished with it without affecting the original.

For training purposes, we will be using the REMIS training database. Perform the following steps to access and connect to Discoverer:

1. Click the Discoverer User icon on your desktop to open Discoverer.
2. Enter your assigned username and password, and the correct database name: (Figure 1.1)

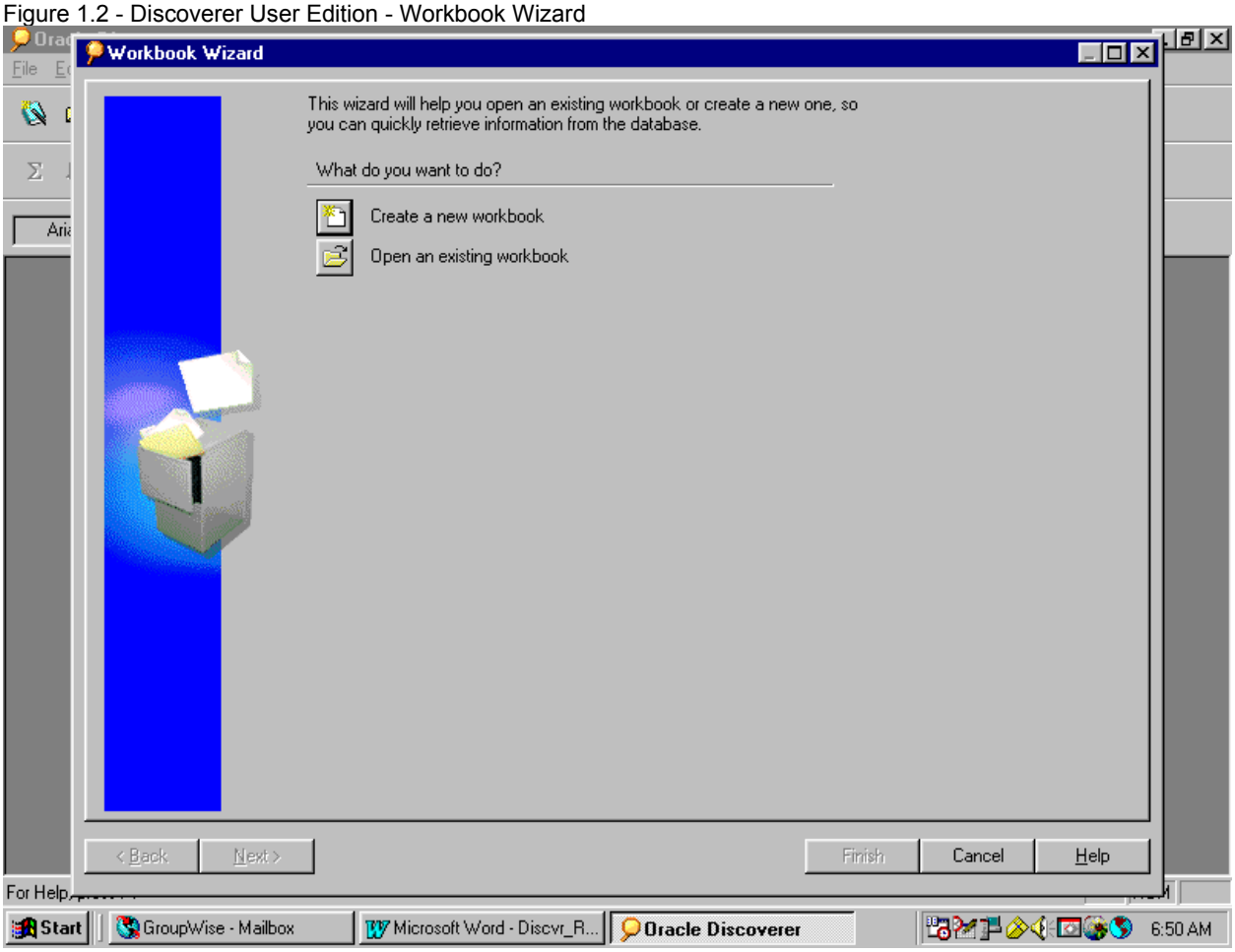
   Username: xxxxxx
   Password: xxxxxx
   Database: remistrg (remisprd for production database)

Figure 1.1 - Discoverer User Edition - Connect to Discoverer



3. Click on the Connect button. A Workbook Wizard page is displayed. (Figure 1.2)

Figure 1.2 - Discoverer User Edition - Workbook Wizard

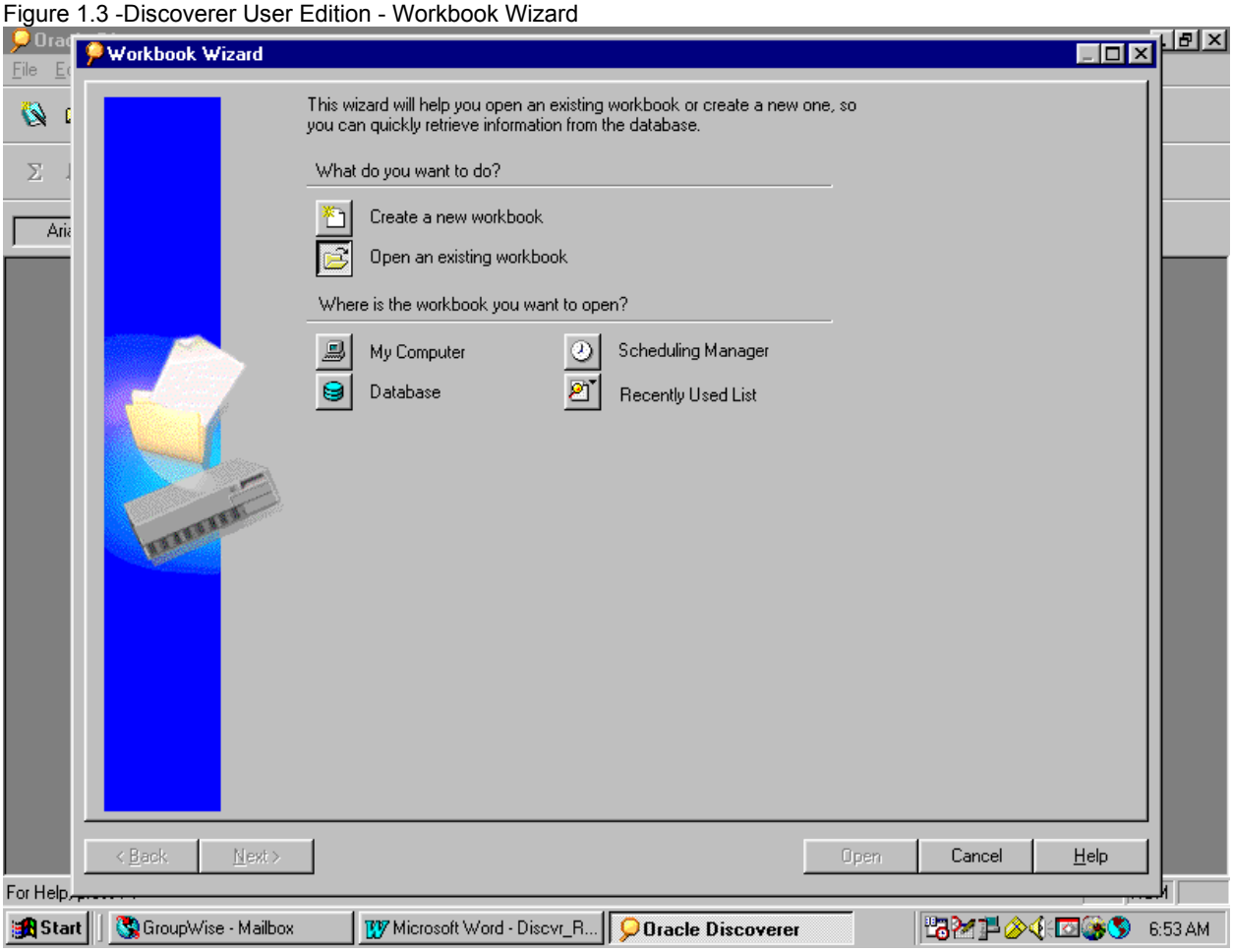


To run a predefined query (workbook):

1. Click on the Open an existing workbook icon.

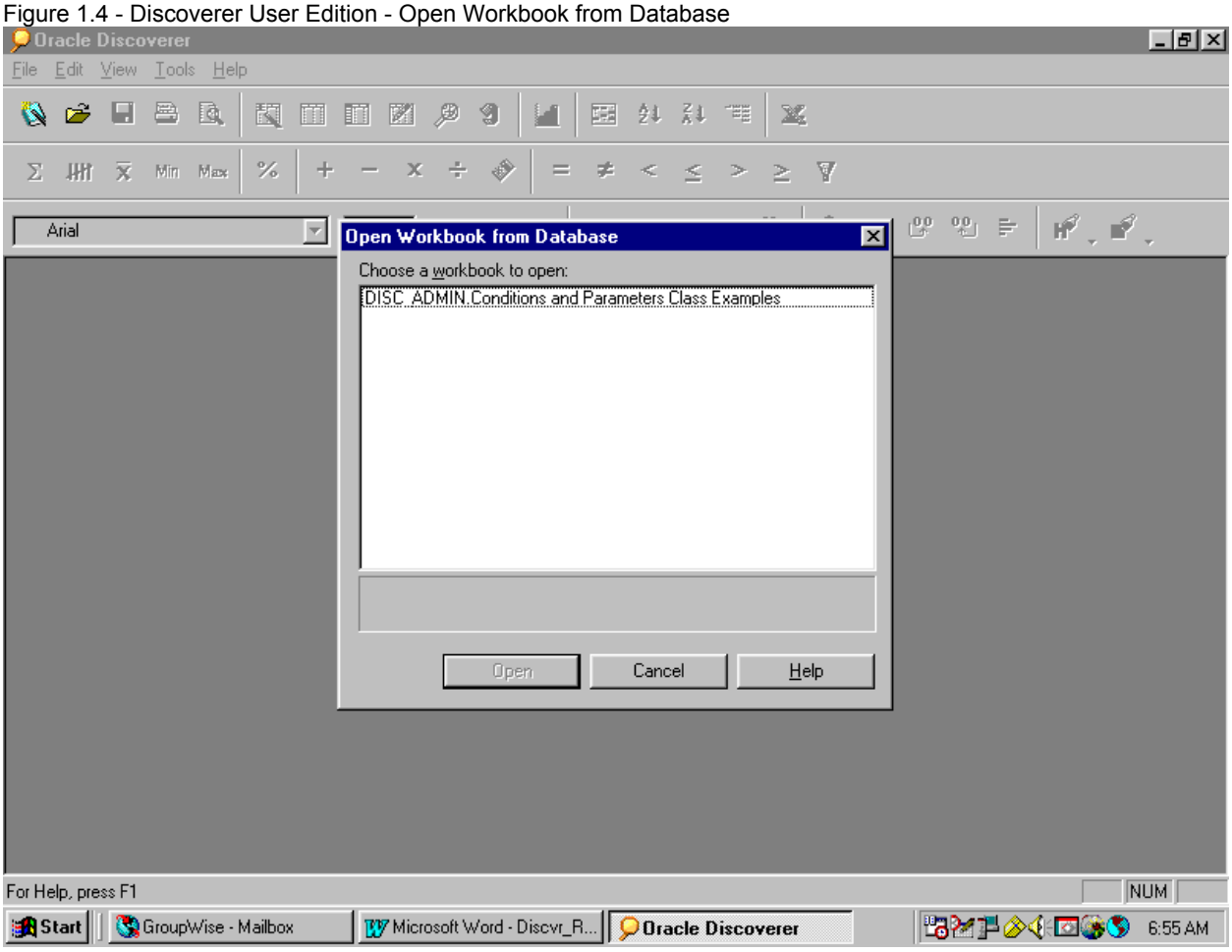
2. At the "Where is the workbook you want to open?" prompt, click on the Database icon. (Figure 1.3)

Figure 1.3 -Discoverer User Edition - Workbook Wizard



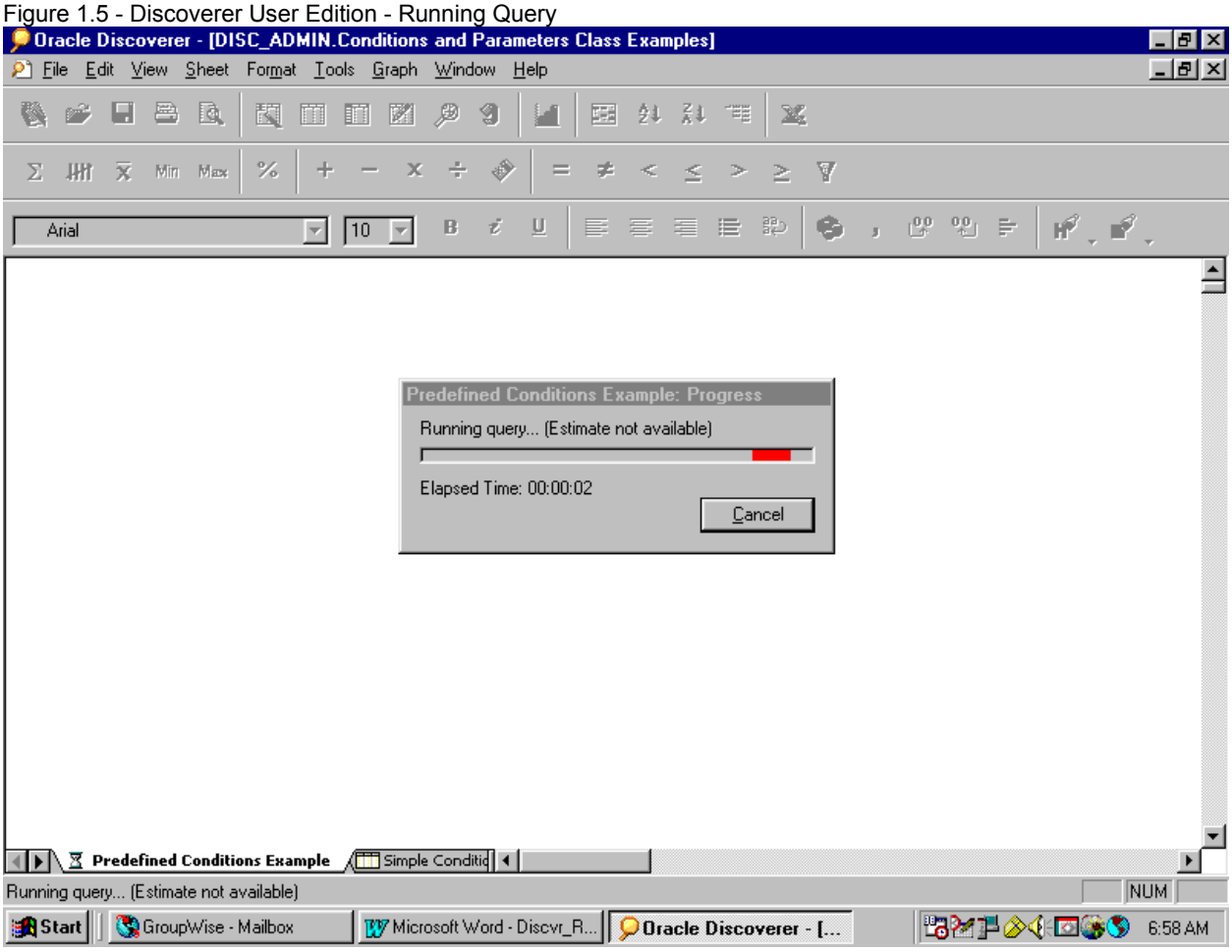
3. An Open Workbook from Database box is displayed with a list of the workbooks currently available to you. All predefined workbooks start with DISC_ADMIN. Any workbooks that you create will not begin with this.
4. Highlight the workbook you want to run by clicking on it, then click on the Open button. (Figure 1.4)

Figure 1.4 - Discoverer User Edition - Open Workbook from Database



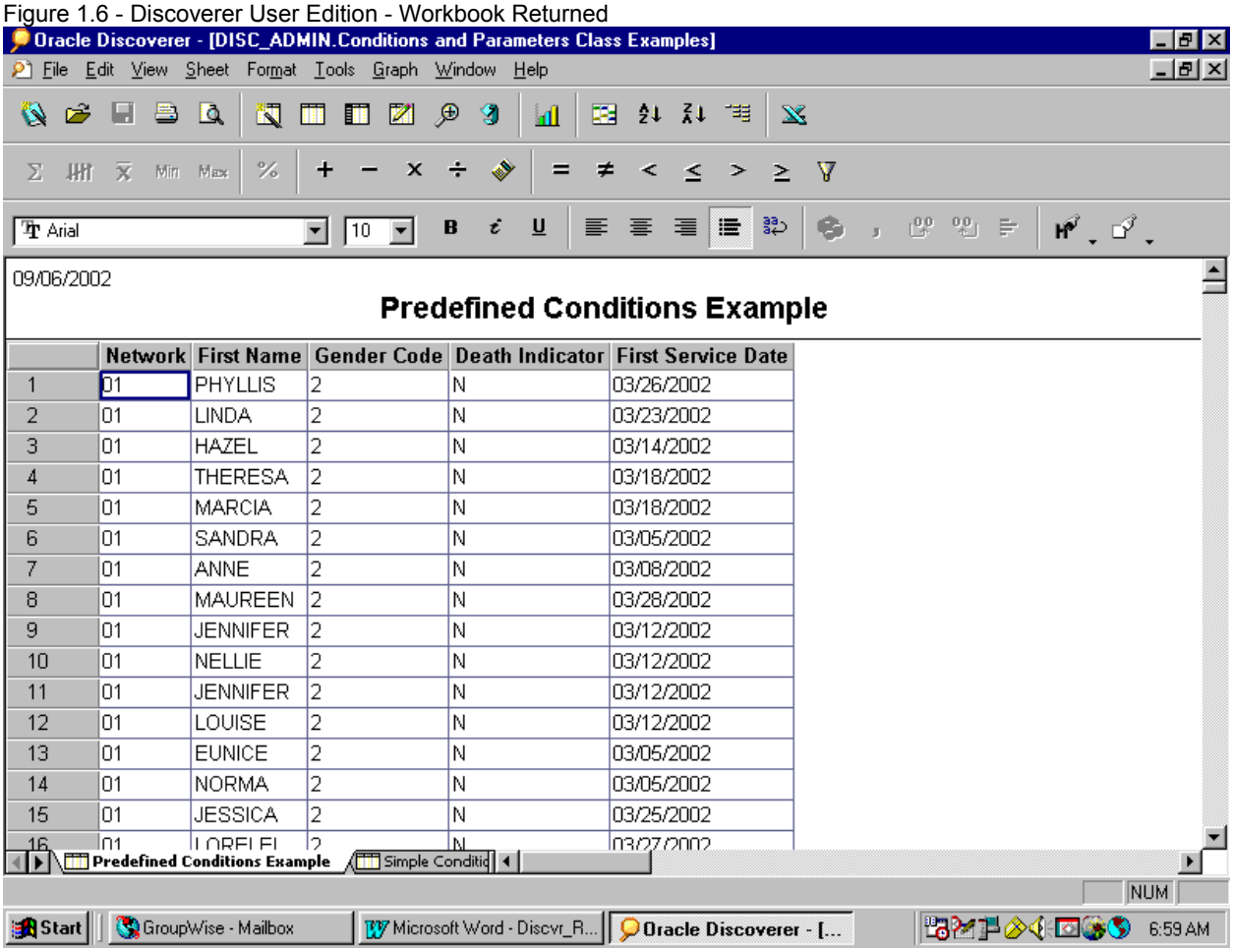
5. A progress box is displayed while the query is running. (Figure 1.5)

Figure 1.5 - Discoverer User Edition - Running Query



6. When completed, the final worksheet (report) is displayed. (Figure 1.6)

Figure 1.6 - Discoverer User Edition - Workbook Returned



	Network	First Name	Gender Code	Death Indicator	First Service Date
1	01	PHYLLIS	2	N	03/26/2002
2	01	LINDA	2	N	03/23/2002
3	01	HAZEL	2	N	03/14/2002
4	01	THERESA	2	N	03/18/2002
5	01	MARCIA	2	N	03/18/2002
6	01	SANDRA	2	N	03/05/2002
7	01	ANNE	2	N	03/08/2002
8	01	MAUREEN	2	N	03/28/2002
9	01	JENNIFER	2	N	03/12/2002
10	01	NELLIE	2	N	03/12/2002
11	01	JENNIFER	2	N	03/12/2002
12	01	LOUISE	2	N	03/12/2002
13	01	EUNICE	2	N	03/05/2002
14	01	NORMA	2	N	03/05/2002
15	01	JESSICA	2	N	03/25/2002
16	01	LORELEI	2	N	03/27/2002

7. To close the workbook, select File|Close. If any changes have been made to the workbook, an Oracle Discoverer box is displayed asking if you want to save the changes to the workbook.
8. Select No to close the workbook.
   OR
   Select Yes to save the workbook. A Save Workbook to Database box is displayed with a list of the workbooks that have been created by the user ID with which you signed onto the database.
   NOTE: When the workbook is saved, the query parameters you selected for the report will be saved, as well as any changes made to the workbook.
14. Enter a name for the workbook in the New Name field.
15. Select Save. The workbook is saved to the database.
    NOTE: Workbooks are accessible only to the user ID who created them, as well as any other user IDs they have been shared with.

Lab Exercise 1

In this lab exercise, you will practice accessing and logging into Discoverer.

1. Click the Discoverer User icon on your desktop to open Discoverer.
2. At the prompt, type in your assigned username, password, and database connection string. The Workbook Wizard window displays.

Topic 2 – Overview of Conditions

What is a condition? We have all set conditions at one time or another, whether in our personal experiences or in our business environments. Webster's New Collegiate Dictionary (copyright 1973 by G. & C. Merriam Co.) provides one definition of the word condition as "3a: a restricting or modifying factor: qualification". The same is true for applying conditions to Discoverer queries. Condition statements provide a means for communicating to the database just what restrictions you want to put on the data that gets returned to you from a query. When you convert your business rules into condition statements, you will get only the data you are expecting to see. Conditions in Discoverer may range from very simple conditions, to complex conditions, to grouped complex conditions, to conditions set at run-time through parameters. This topic will address each of those types of conditions.

Simple conditions are those conditions that define a rule about one piece of information (column) and the value or values it may contain in order to be selected for the query. An example of a simple condition is that only data for a particular ESRD Network will be included in the query. This is accomplished by adding the condition

Network = '01'

to the query. Conversely, in order to include data for all Networks except a particular Network, you would add the condition

Network <> '01'

to the query. You can code any number of simple conditions in a query. When you do so, the query assumes that ALL of those conditions must be met in order for the data to be returned. It inserts an implied AND between each condition statement. Be careful that you do not include contradictory conditions in your query definitions, or you will likely receive unexpected results (no data).

Complex conditions take simple conditions to the next level. Complex conditions allow you to combine two or more simple conditions into one condition statement for the query. Complex conditions are useful when you have a business need that requires an either/or situation. For example, if you wanted to include data in a query for either Network 1 or Network 2, you could combine that either/or requirement into one complex condition:

Network = '01' OR
Network = '02'

As you become familiar with all of the operator options available in conditions, you will notice that some simple either/or situations can be handled with a simple condition statement. In the example above, the condition statement:

Network IN ('01', '02')

produces the same results as coding it as a complex condition. In those instances, it is completely up to individual preference.

Grouped complex conditions are an extremely useful albeit complicated method of incorporating business needs into a query. These condition statements allow you to mix both 'and' conditions and 'either/or' conditions, or any multiple statements of either, at any number of levels, all in one statement. Definitely not for the faint at heart! These conditions are useful in those situations where you want to retrieve only one report of the same data items, but the business requirements reflect multiple subsets. For example, you want to create a report showing selected patients in Network 1 whose most recent transplant was on or after 01/01/2002 in the same report with selected patients in Network 2 whose most recent transplant was on or after 03/01/2002. You would need to code this as a grouped complex condition:

Network = '01' AND
Most Recent Transplant Date >= '01/01/2002'
OR
Network = '02' AND
Most Recent Transplant Date >= '03/01/2002'

Looks simple enough, doesn't it? There are a few tricks in entering these grouped complex conditions that you will want to remember. We will go over them in more detail in a later topic.

Sometimes the same data items are needed in a report that is run frequently, but the condition values may change each time the report is run. You can accomplish this in one of two ways. You can edit your query every time you want to run the report, changing the condition to reflect the new values you're looking for. This can become very tedious over time, especially if you are sharing this report with other people who may not appreciate their report changing every once in a while for some unknown reason! The other way to make a report reflect changing business rules is to make it flexible by creating *parameterized conditions*. These conditions do not contain the actual value or values used to restrict the report. Instead, they contain placeholders, or parameters, that hold the place in the query for a value to be inserted at runtime. The parameter boxes automatically pop up, requiring entry of the values for one or more conditions, prior to Discoverer running the query. If you are creating queries that you want to run repeatedly but with different values for the same conditions, then parameters are the way to go. They are relatively easy to set up, and will make the report much more flexible and user-friendly.

Topic 3 – Select Predefined Conditions

First-time Discoverer users are likely to look at queries that have already been prepared by someone else to get familiar with the environment or to get ideas for how to construct a query on their own.

A few points of clarification are in order at this time:
1. The REMIS Reporting business area found in Discoverer has been predefined. A business area is a set of related information with a common purpose.
2. Within a business area there are multiple folders, which represent database tables. Within each folder there are multiple items, which represent columns on the table.
3. A few workbooks have already been created by IFMC to be used as a starting point for you to begin building your own queries in Discoverer.
4. When you open a workbook owned by someone else's user id, you will not be working with the original workbook. Instead, you are working with a copy of that workbook. You may save your copy when you are finished with it without affecting the original.

Conditions in existing worksheets are easily turned on or off. This makes a query flexible in that you can retrieve the same items of data, first selectively choosing by the condition you set, then turning off that condition and retrieving the entire set of data, without having to recreate the query. For example, your business need is to retrieve data items for all female patients whose first service date was after 3/1/2002, first for a particular Network only, then for all Networks. Conditions are set up as follows:

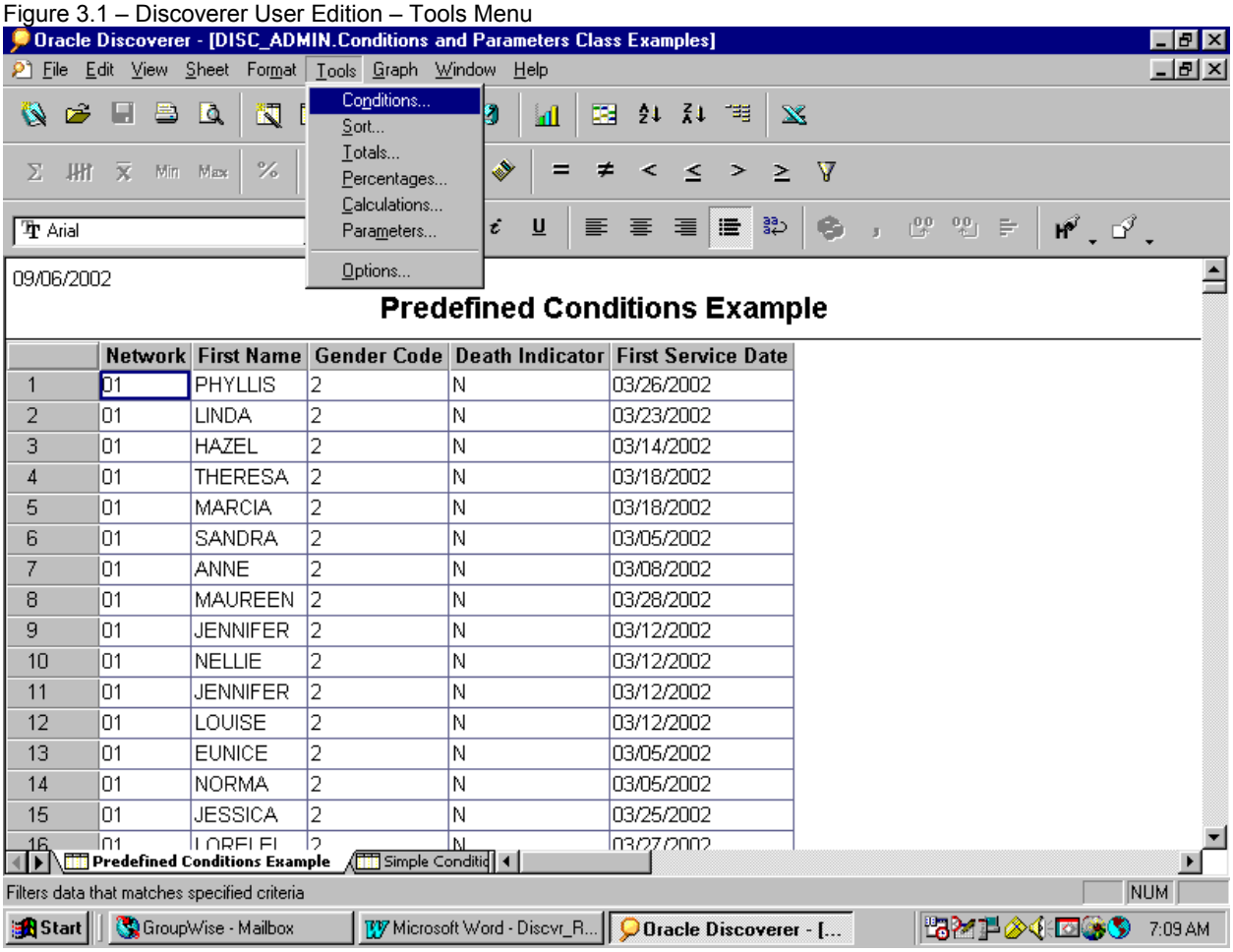
Gender = 'Female' AND
First Service Date > '03/01/2002' AND
Network = '01'

By simply turning off the Network condition from the query, the worksheet will now retrieve all female patients with first service dates after 03/01/2002 for all Networks. By turning off the First Service Date condition, the worksheet will retrieve all female patients for Network 01, regardless of when their first service date occurred. By turning off the Gender condition, the worksheet will retrieve all patients with first service dates after 03/01/2002. As you can see, the possibilities are endless. All of these queries can be accomplished with only one worksheet definition and the click of a checkbox.

To turn a condition on or off, follow these steps:

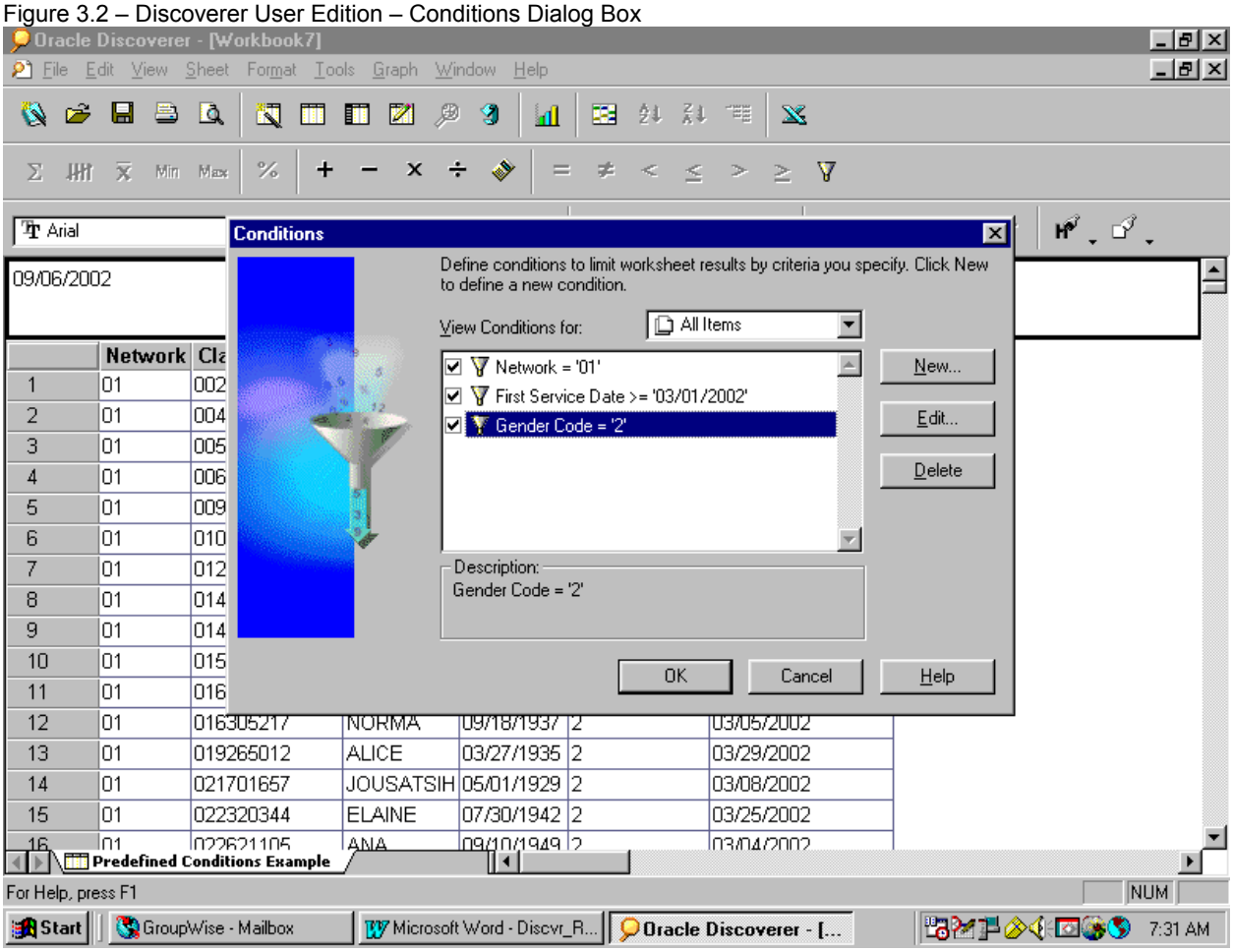
1. Open the worksheet containing the query you want to run.
2. Click the Tools menu item, and select Conditions… from the drop down menu. (Figure 3.1)

Figure 3.1 – Discoverer User Edition – Tools Menu



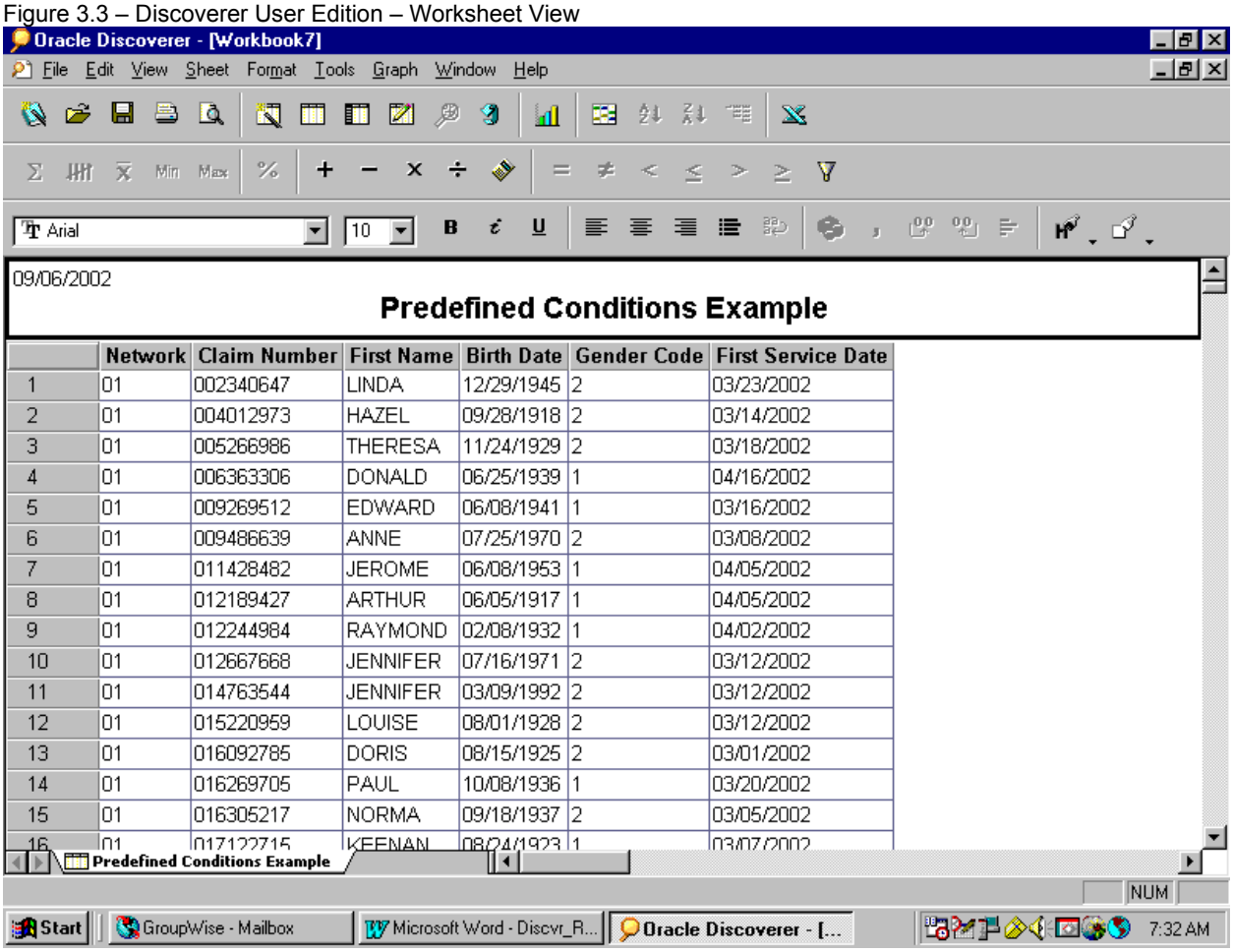
3. Locate the condition to be turned on or off, and click in the box to the left of the condition to set it to the desired status: a check mark in the box indicates the condition is turned on; a blank box indicates the condition is turned off. (Figure 3.2)

Figure 3.2 – Discoverer User Edition – Conditions Dialog Box



4. Click OK to return to the worksheet.
5. The query will retrieve the data according to the new condition settings. (Figure 3.3)

Figure 3.3 – Discoverer User Edition – Worksheet View



	Network	Claim Number	First Name	Birth Date	Gender Code	First Service Date
1	01	002340647	LINDA	12/29/1945	2	03/23/2002
2	01	004012973	HAZEL	09/28/1918	2	03/14/2002
3	01	005266986	THERESA	11/24/1929	2	03/18/2002
4	01	006363306	DONALD	06/25/1939	1	04/16/2002
5	01	009269512	EDWARD	06/08/1941	1	03/16/2002
6	01	009486639	ANNE	07/25/1970	2	03/08/2002
7	01	011428482	JEROME	06/08/1953	1	04/05/2002
8	01	012189427	ARTHUR	06/05/1917	1	04/05/2002
9	01	012244984	RAYMOND	02/08/1932	1	04/02/2002
10	01	012667668	JENNIFER	07/16/1971	2	03/12/2002
11	01	014763544	JENNIFER	03/09/1992	2	03/12/2002
12	01	015220959	LOUISE	08/01/1928	2	03/12/2002
13	01	016092785	DORIS	08/15/1925	2	03/01/2002
14	01	016269705	PAUL	10/08/1936	1	03/20/2002
15	01	016305217	NORMA	09/18/1937	2	03/05/2002
16	01	017122715	KEENAN	08/24/1923	1	03/07/2002

Note: If you have a column on your report that shows the word NULL in the column, you may change the options setting for your Discoverer session. Access the Tools menu item, select Options, and then go to the Sheet Format tab page. On that page you may set the value to display for Null values. (For non-programmers, a Null value means simply there is no data in the item for that row.)

Lab Exercise 2

In this lab exercise, you will practice opening a predefined workbook, then selecting and deselecting existing conditions to retrieve different result sets.

1. Login to Discoverer. When the Workbook Wizard screen displays, click the “Open an existing workbook” icon.
2. Answer the question, “Where is the workbook you want to open?” by clicking the Database button.
3. On the Open Workbook from Database screen, select the Conditions and Parameters Class Labs workbook and click the Open button to open it.
4. At the prompt, Do you want to run the query for the sheet “Predefined Conditions Lab 1”?, click Yes.
5. The query will run. How many patients are listed?
6. Access the Conditions for the worksheet by clicking on the Tools menu item, then selecting Conditions from the dropdown menu.
7. Check the box to the left of the condition, “Gender = ‘Female’". Click OK.
8. The query will re-retrieve the data. Now how many patients are listed?
9. Save the workbook by clicking on the File menu item then selecting Save As from the dropdown. The Save Workbook box displays. Click on Save. Leave the Conditions and Parameters Class Labs workbook open.

Topic 4 – Create Simple Conditions

This topic will outline the steps to follow in defining a query with conditions.

Step 1. Define your business need.
Before creating a query with a condition to restrict the data to be returned, compose a statement of your business need. What pieces of information will I need to look at? What groups of data am I interested in looking at? Is there any data that I specifically do or don't want to see? An example of a statement of business need for a REMIS query is:

"I need a list of all the patients in Network 4 who have had transplants in 2002. For each patient, I need to see their Network, claim number, name, date of birth, gender, and date of the transplant."

From this simple business statement, you can start to define your query with conditions.

Step 2. List pieces of information to be included on the final query result set.
Using your statement of business need, identify the elements to include in the report. In our example, the elements that should be on this list are printed in bold type:

*"I need a list of all the patients in Network 4 who have had transplants in 2002. For each patient, I need to see their **Network**, **claim number**, **name**, **date of birth**, **gender**, and **date** of the transplant."*

In short, the report should include Network, Claim Number, Name, Birth Date, Gender, and Transplant Date.

Step 3. List pieces of information to be used to filter in/out data from the final query result set.
Again, using your statement of business need, isolate the elements that identify the restrictions you want to place on the data. These elements will be the basis of your condition statements in the query. In our example, these condition elements are identified in bold type:

*"I need a list of all the patients in **Network 4** who have had transplants in **2002**. For each patient, I need to see their Network, claim number, name, date of birth, gender, and date of the transplant."*

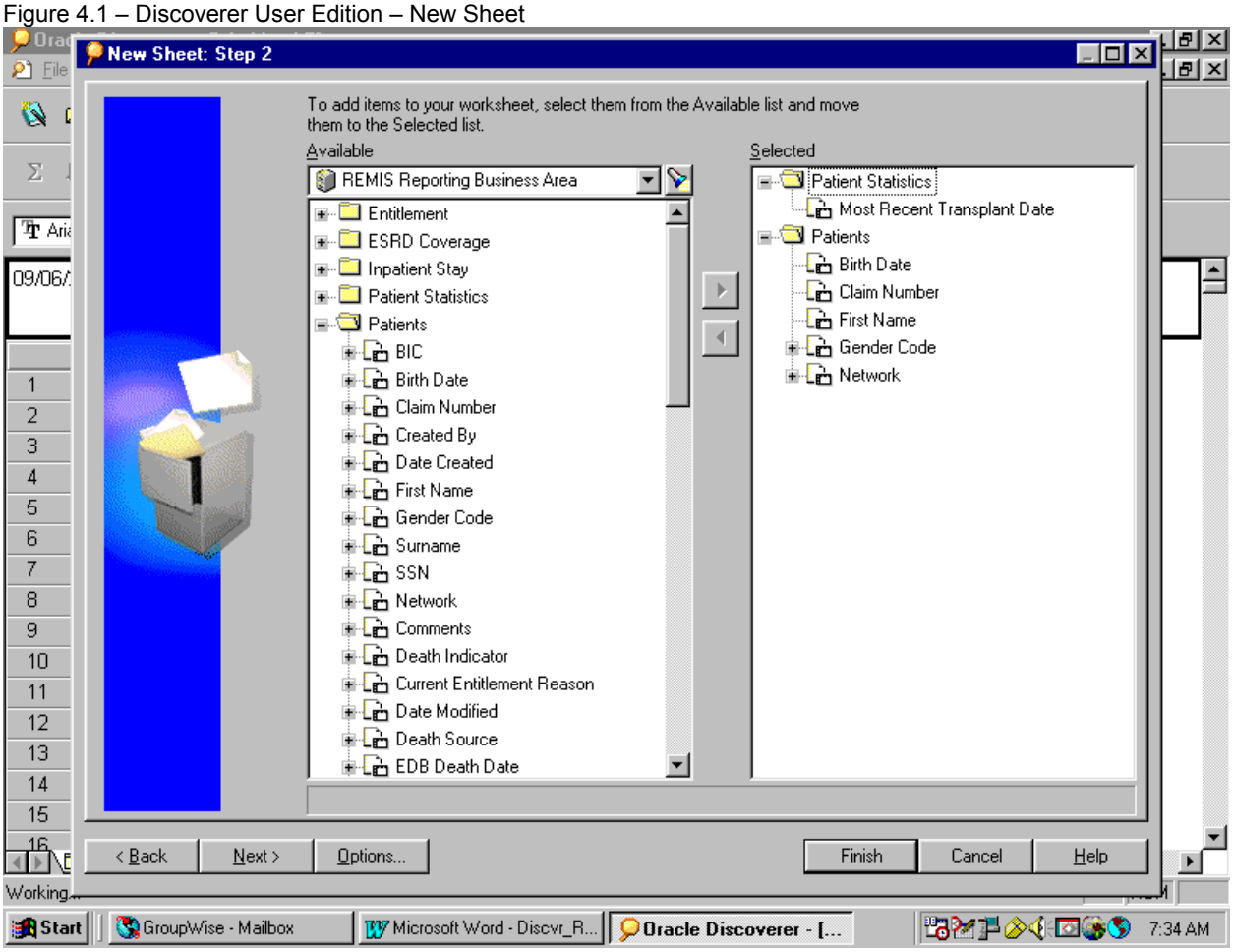
The conditions for this query should limit the data retrieved into the result set to Network 4 patients only, for the year 2002 only.

Step 4. Build your query.

Now that we have identified the items to include in the final report, and we know what limits we want to place on the data, we can begin to build the query. Access Discoverer and open an existing workbook, or create a new one. To create a new query in an existing workbook, add a new worksheet to the workbook. When you create a new workbook, Discoverer automatically inserts a new worksheet into the new workbook.

For this example, a new worksheet has been added to the existing Conditions and Parameters Class Examples workbook. Access the REMIS Reporting business area. Begin to create the new query in Discoverer by choosing the Network, Claim Number, First Name, Birth Date, and Gender Code from the Patient folder, and Most Recent Transplant Date from the Patient Statistics folder. (Figure 4.1)

Figure 4.1 – Discoverer User Edition – New Sheet



Continue building the worksheet by following the instructions in the wizard. Navigate between the wizard steps by using the Next and Back buttons. Use the layout page to rearrange the order of the columns on the report or to add/remove page items.

Conditions are added on step 4 of the wizard. Try to think of a condition statement in a query as a mathematics equation. Instead of numbers and an assumed equal sign, the condition statement defines the relationship between an item in the database and a specific value or set of values using one of several operators. Examples of operators that may be used are:

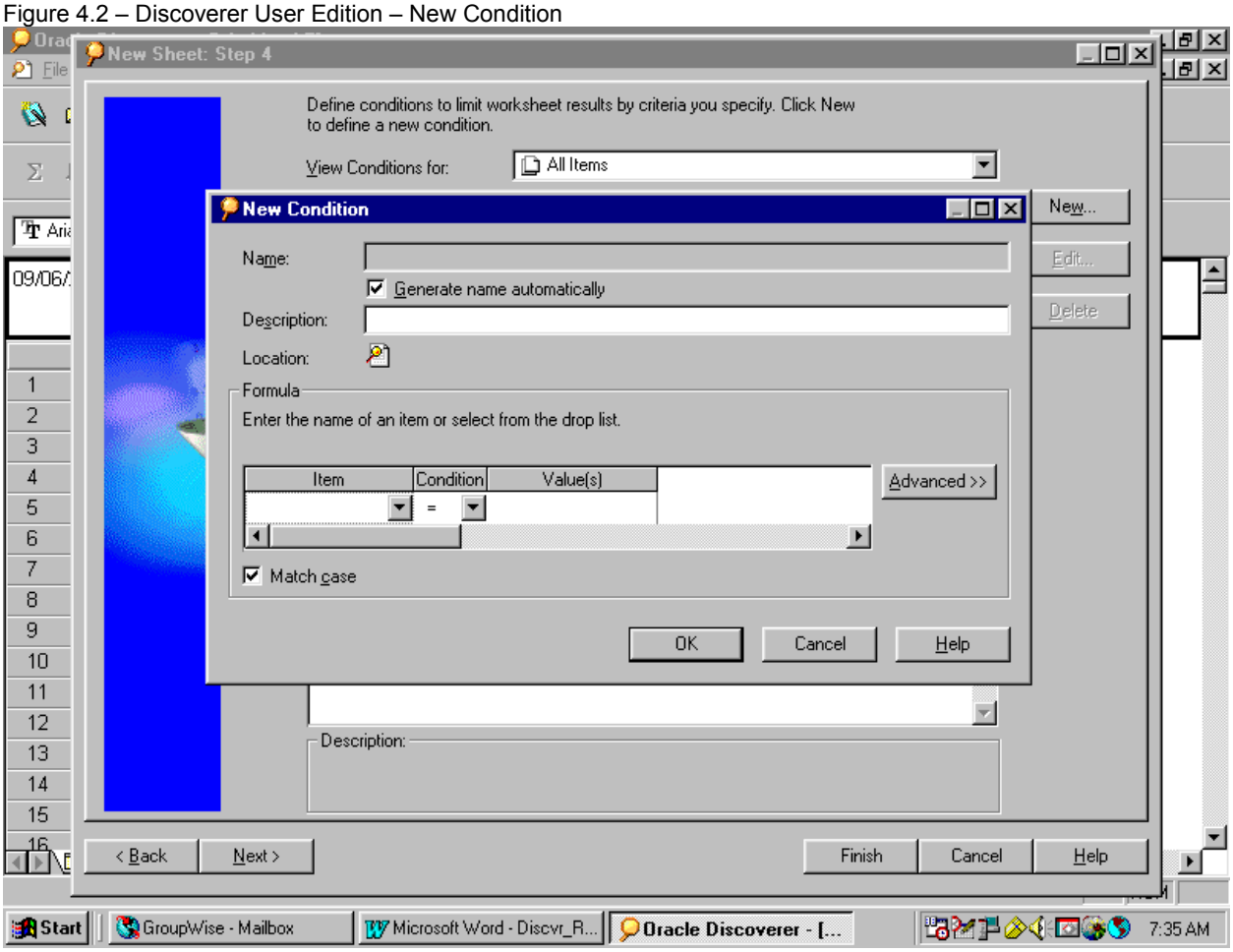
Table 4.1 – Condition Operator options

Operator	Description	Operator	Description
=	Equal To	<>	Not Equal To
>	Greater Than	<	Less Than
>=	Greater Than or Equal To	<=	Less Than or Equal To
LIKE	Similar To; item value matches partial text string with wild card characters	NOT LIKE	Not Similar To; item value doesn't match partial text string with wild card characters
IN	Contained In; item value is found in the specified list.	NOT IN	Not Contained In; item value is not found in the specified list.
IS NULL	Empty; no data in the item. This is not the same as a space or a zero.	IS NOT NULL	Not Empty; there is data in the item.
BETWEEN	In Range of Values; both ends of specified range are included	NOT BETWEEN	Not In Range of Values; both ends of specified range are excluded
!=	Not Equal To (same as <>)	^=	Not Equal To (same as <>)

To add a new condition, click the New button on the right side of the page. The New Condition window opens. (Figure 4.2)

The main items on the New Condition window include the condition's Name, Description, and Formula. When the 'Generate name automatically' box is checked, the Name box is protected from entry. When this check box is left unchecked, the Name box is unprotected, and you may enter in your own name for the condition. The Description field is optional, and provides an area to enter information about the condition. (Note: The Description displays on the main Conditions page in the area below the list of available conditions for the workbook. If no description is added, the condition formula is displayed in the description area instead.) The Formula box is required, and is where you specify the item, the operator, and the specific value or set of values to define the condition. When you leave the 'Generate name automatically' box checked, Discoverer uses the items selected in the Formula to generate the name of the condition.

Figure 4.2 – Discoverer User Edition – New Condition

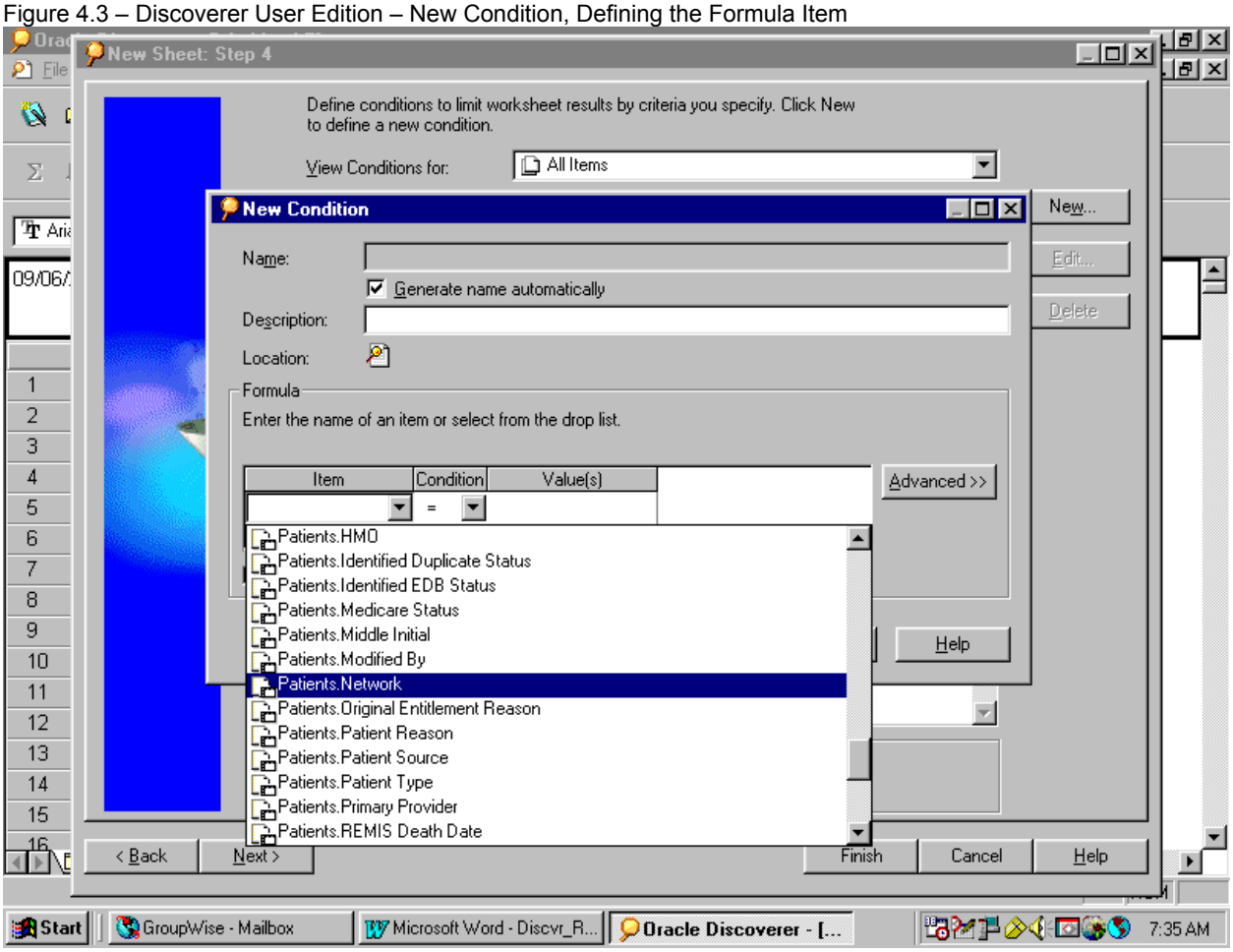


The first entry in the Formula box is Item. (Figure 4.3) Click the down arrow to display the list of available options to select. Available options are a list of all of the items from folders that were used to build the initial result set of the query. Also in the list are options for creating a calculation or select another condition to be used in place of an item from a folder. We won't be discussing the calculation option in this class, but you can read more about them in Discoverer's online User Guide or in the online help. Selecting another condition will be addressed during the lab exercise for this topic.

For the purposes of this class, we will select the item, Patient.Network. This item fulfills the first identified filter requirement from Step 3, '**Network 4'**.

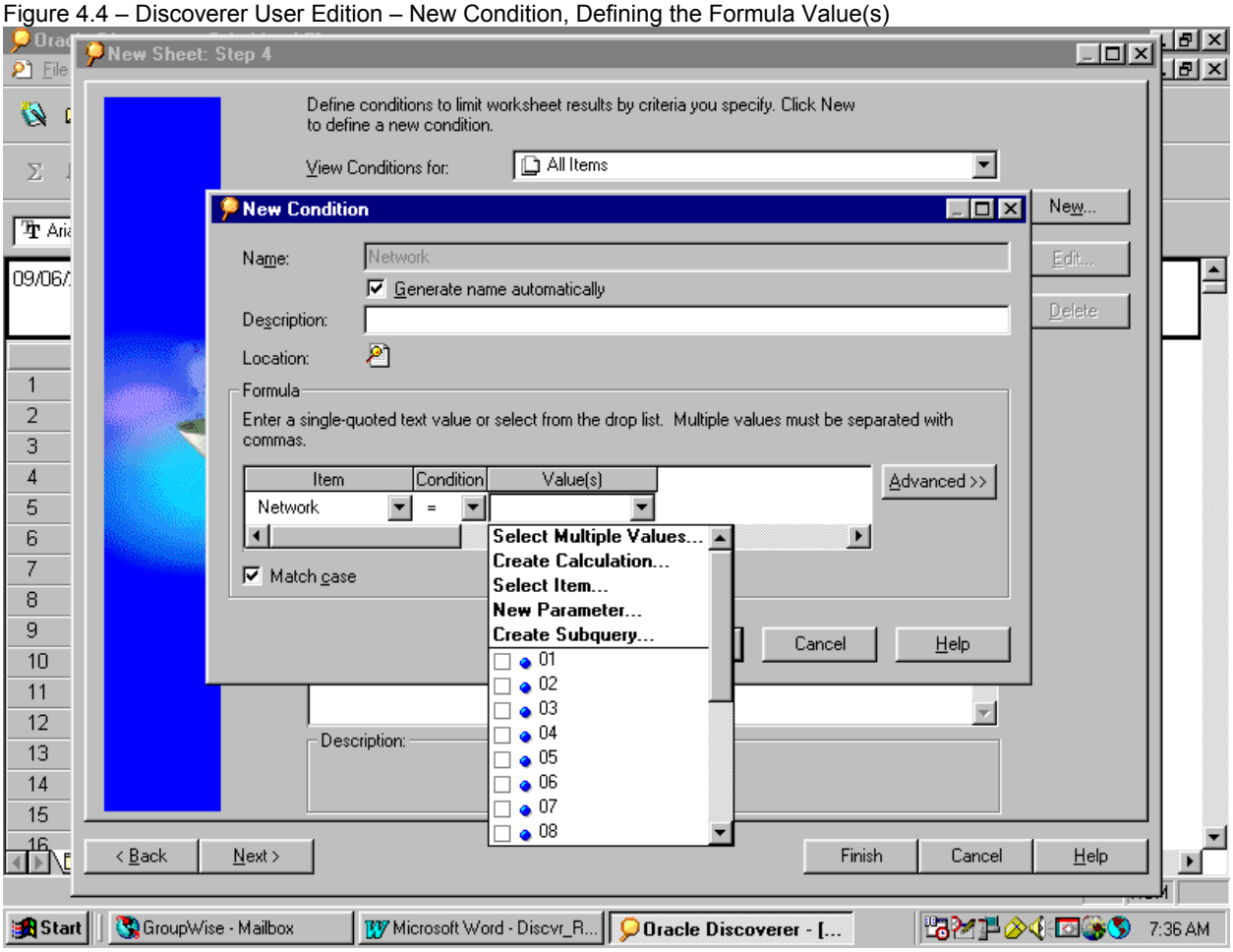
The Condition operator automatically defaults to '='. To see the remaining options identified in Table 3.1, click the down arrow in the Condition box. For this condition, we want the default '=' so we will move on to the Value(s) box.

Figure 4.3 – Discoverer User Edition – New Condition, Defining the Formula Item



The last options in the Formula box are the Value(s) options (Figure 4.4). Click the down arrow to display the list of available options to select. If the Item selected for the condition has been defined in the Business Area with a List of Values associated with it, that list will display at the bottom of this dropdown box. To select just one value from this list, click the box to the left of the value to select it.

Figure 4.4 – Discoverer User Edition – New Condition, Defining the Formula Value(s)



Other options in the Value(s) dropdown are available to add flexibility when building a condition. Select Multiple Values… will allow you to select more than one of the list of values available for the condition item if you want to include several values for that item in the query. This is a useful alternative to coding complex conditions to achieve an either/or situation for multiple values for the same item. (Figure 4.5)

For example, in one query, you want to return data for both Network 4 and Network 5 patients. To code this as a complex condition, you would need to create the condition with two parts:

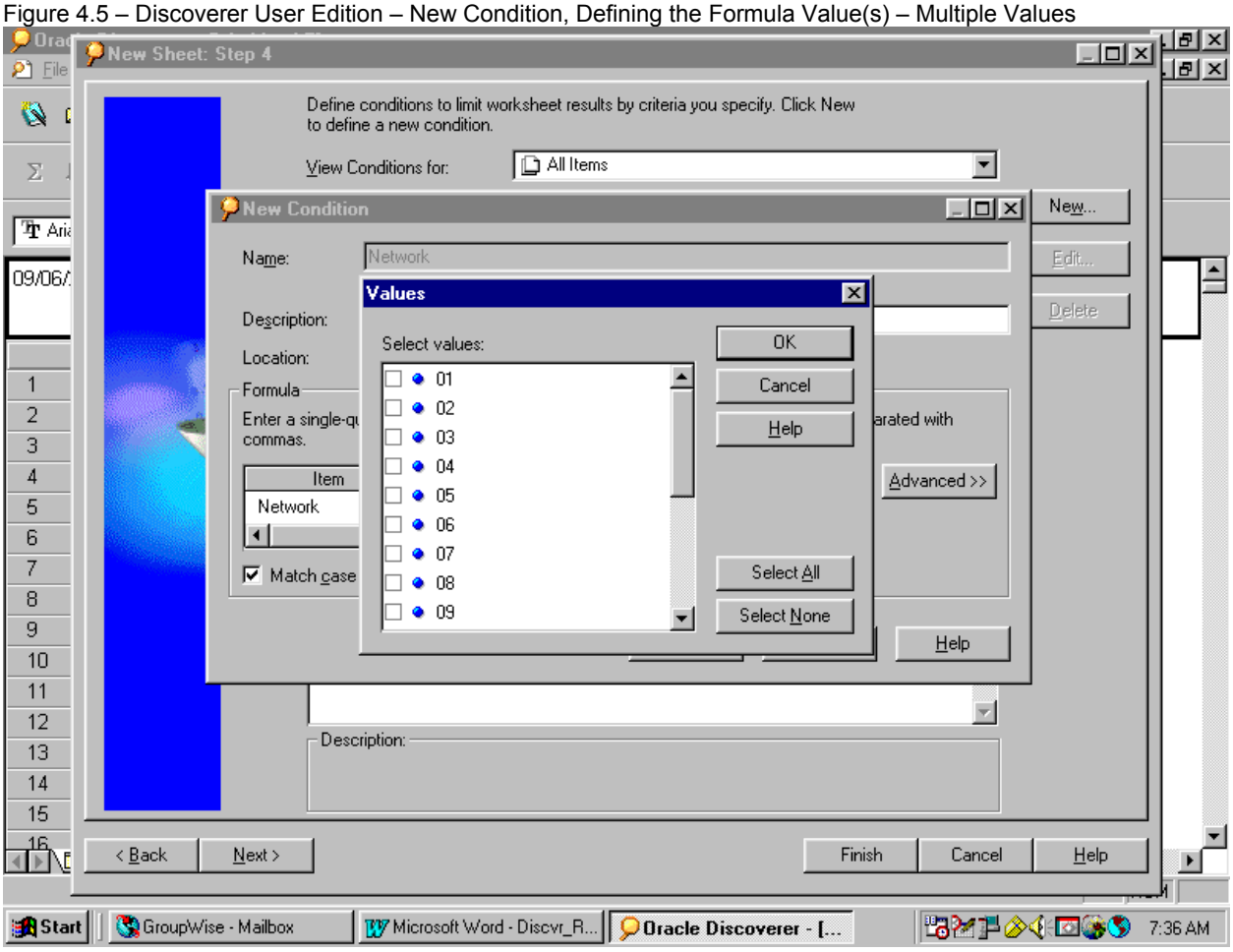
Network = '4' OR
Network = '5'

Instead, use the multiple values list and select both values ('4', '5') to be included in a simple condition to achieve the same results:

Network IN ('4', '5')

When selecting multiple values for a condition, remember to use an operator value of 'IN'.

Figure 4.5 – Discoverer User Edition – New Condition, Defining the Formula Value(s) – Multiple Values

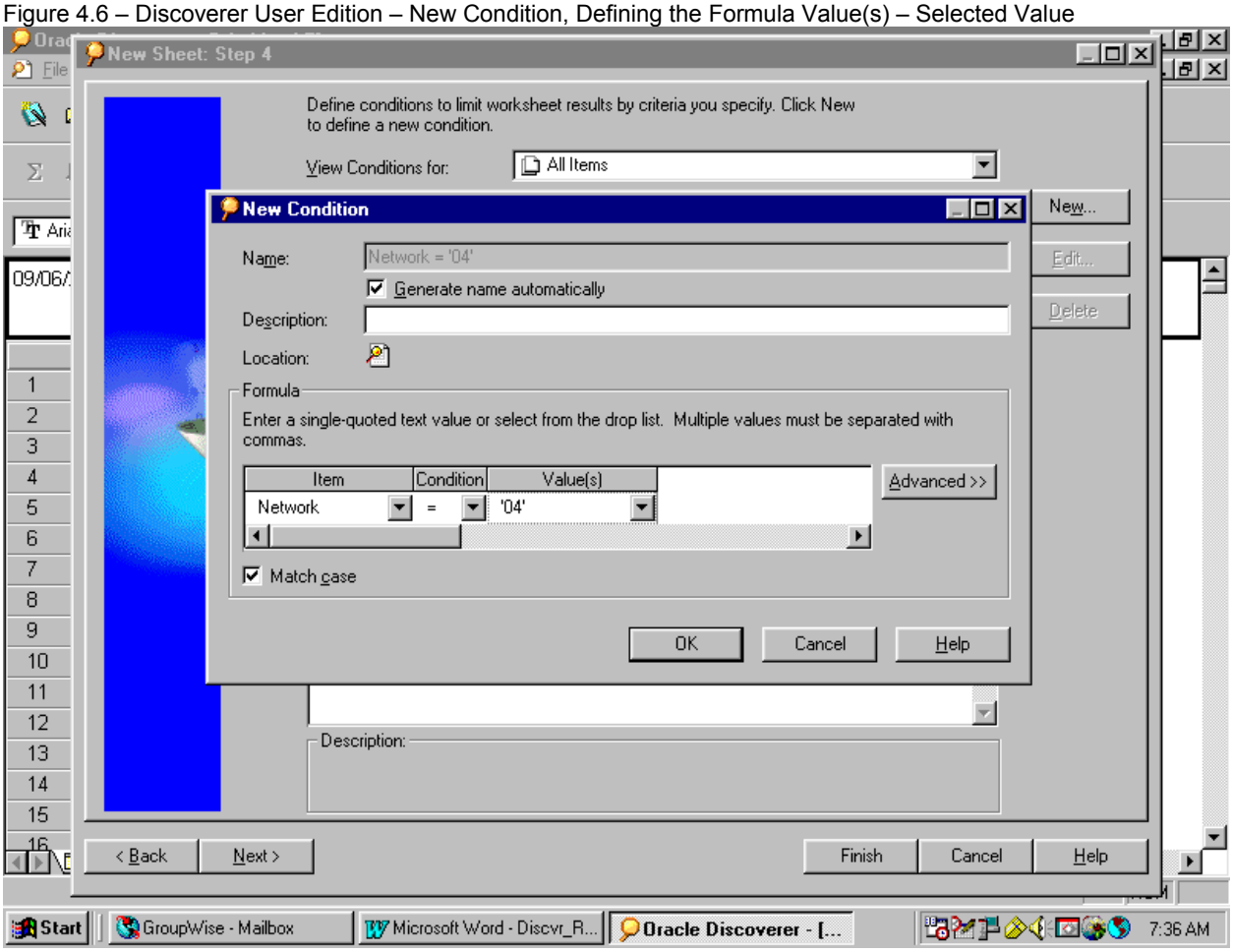


Other options for selecting the value for the condition include creating calculations, selecting another item from the same or any other folder referenced in the query, adding a parameter value, or creating a subquery. We will cover creating conditions with parameters in another topic.

For the purposes of this topic, we will select just the value of '4', for Network 4. This item fulfills the value for the first identified filter requirement from Step 3, '**Network 4**'. (Figure 4.6)

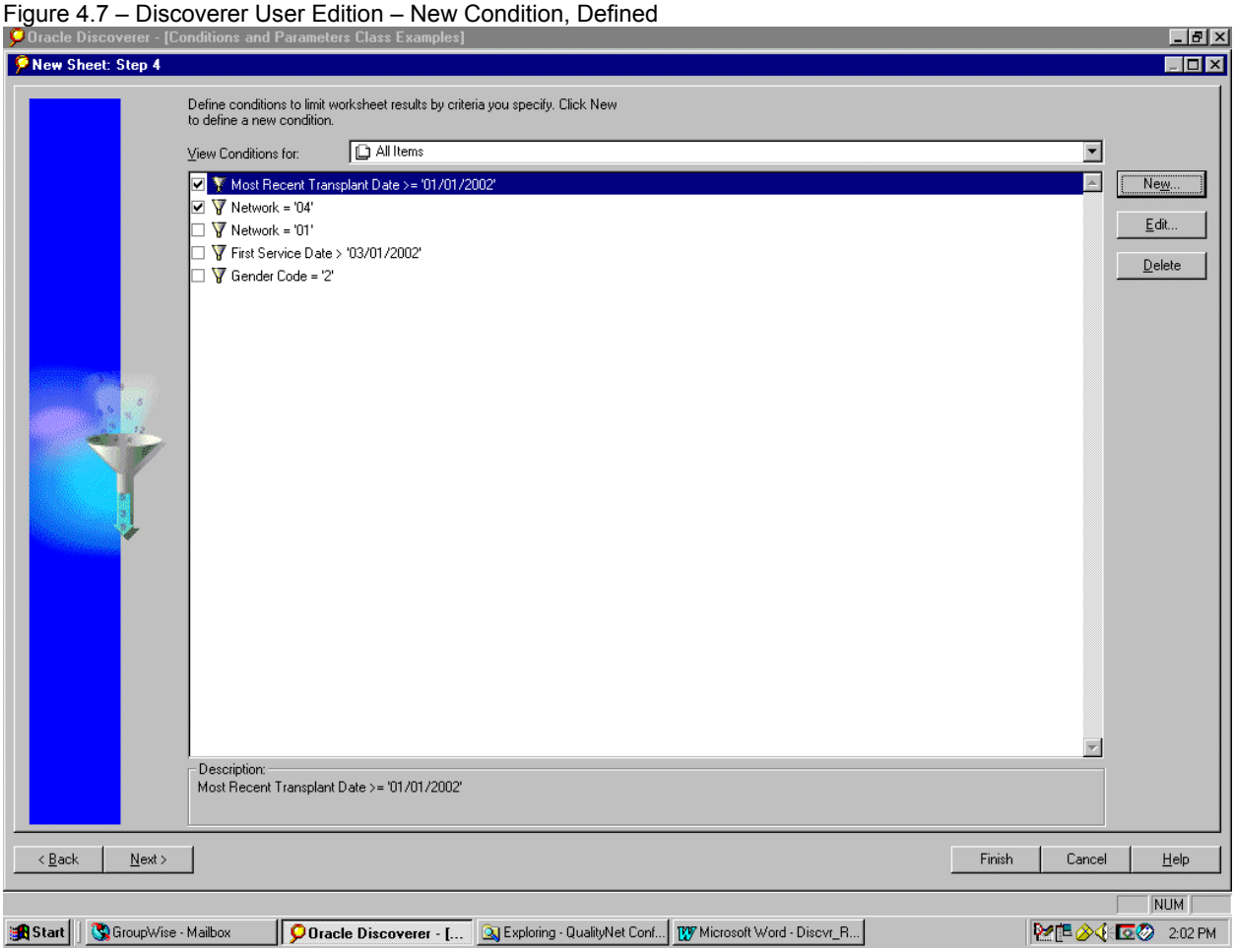
The last check box to mention is the 'Match case' box. When this box is checked, the query will force the condition to match the value exactly as it is typed in the formula Value(s). When this box is unchecked, the query will actually look for either an upper or lower case character version of what is listed in the formula Value(s).

Figure 4.6 – Discoverer User Edition – New Condition, Defining the Formula Value(s) – Selected Value



When the condition has been fully defined, click OK to return to the condition list. Notice that the condition is automatically checked so that it is active for the query. (Figure 4.7) Continue building the specifications for the new sheet. When you are finished, click Finish to close the New Wizard and execute the query.

Figure 4.7 – Discoverer User Edition – New Condition, Defined



Lab Exercise 3

In this lab exercise, you will practice identifying the elements to be included in the query result set and the items used for conditions to filter the data returned from the provided statement of business need.

Statement of Business Need:

I need a list of all living patients in Network 4 who have had a kidney transplant fail sometime during 2002. For each patient, I want to see their Network, claim number, name, gender, birth date, date of transplant, and the date the transplant failed.

1. List the elements that should be included in the final result set for the query to fill this business need.

2. List the items that should be used to create conditions to restrict the data returned in the final result set.

Lab Exercise 4

In this lab exercise, you will use the elements and conditions defined in Lab Exercise 2 to build a new sheet in the Conditions and Parameters Class Labs workbook you should already have open.

1. Click the Sheet menu item, and select the New Sheet option from the dropdown list.
2. The New Sheet window opens. Select the default Table results display format and click Next to continue.
3. Verify that the REMIS business area is active for the sheet.
4. Open the Patient folder by clicking on the + to the left of the folder name and icon.
5. From the Patient folder, select the Network, Claim Number, First Name, Gender, and Birth Date, individually or in a group (press the Ctrl key as you click the items). Add them to the Selected items list by clicking on the right arrow button between the two white boxes.

   (This satisfies the Network, claim number, name, gender, and birth date items from the list of elements to be selected for the query.)
6. Close the Patient folder by clicking on the – to the left of the folder name and icon.
7. Open the Transplant folder by clicking on the + to the left of the folder name and icon. Use the vertical scroll bar to view all the items available in the folder.
8. From the Transplant folder, select Transplant Date and add to the Selected items list.
9. Click Next to continue. Step 3 enables the layout of the report to be changed. Arrange it as you see fit, then click Next to continue.
10. Create a new condition by clicking on the New… button to the right of the condition list box.
11. Leave the ‘Generate name automatically’ box checked.
12. Enter ‘Network = 04" into the Description field.
13. Select “Patient.Network" from the Item dropdown box.
14. Leave the Condition operator defaulted to =.
15. Select ’04’ from the Value(s) dropdown box.
16. Leave the ‘Match case’ box checked.
17. Click OK to continue.
18. The new ‘Network = 04’ condition has been added and is automatically checked to make it active.

    (This satisfies the first of the filtering requirements for the query.)
19. Create another new condition by clicking on the New… button.
20. Uncheck the ‘Generate name automatically’ box.
21. Enter ‘Transplant dates in 2002' into the Name field.
22. Leave the Description field blank.
23. Select 'Transplant.Transplant Date' from the Item dropdown box.
24. Select the '>=' operator.
25. Enter '01/01/2002' in the Value(s) box.

26. Click OK to continue.
27. The new ‘Transplant dates in 2002’ condition has been added and is automatically checked to make it active.
28. Note the differences in the names and descriptions of the two conditions you just created.
29. Click Next to continue to Step 5.
30. Optional: Add sorting to your query. Click the Add button, then select Transplant Date from the dropdown list to add a sort by transplant date to the query. Click Add again, and select Transplant Failed Date from the dropdown list to add a secondary sort by transplant failed date to the query.
31. Click Next to continue to Step 6. We will not need any calculations for this query, so click Finish to conclude building the query definition.
32. The New Sheet Wizard closes, and Discoverer runs the query. How many rows were returned?
33. Rename the sheet by double-clicking on the sheet tab, or by right-clicking on the sheet tab and selecting the Rename Sheet option, or by clicking on the Sheet menu item then selecting Rename Sheet. The Rename Sheet box opens. Rename the sheet to ‘Simple Conditions Lab 4’. Click OK to return to the sheet.
34. Optional: Change the title for the sheet by double clicking in the title area. The Edit Title box opens. Replace the default text in the title edit box either manually by typing in the new title, or use one of the Insert options, such as the &Sheet name.
35. Save your workbook in its current state by clicking on the File menu item then selecting Save from the dropdown, or clicking the Save icon on the menu toolbar.

Topic 5 – Create Complex Conditions

This topic will outline the steps to follow in adding complex conditions to a query.

As described in the Overview of Conditions section, there are times when a query requires an either/or decision, or you simply want to contain multiple simple conditions into one. This is the situation where you want to create a complex condition.

Complex conditions and the more involved grouped complex conditions can be created with as little or as much complexity as is needed. A word of advice: The more complicated the condition, the greater chance for coding errors resulting in unexpected query result sets. Start out with simple conditions, then group them together from the inside out. Careful planning will help you to insure valid, expected result sets from the query.

For the examples in this topic, the previous Simple Conditions Example worksheet was copied to create a new sheet in the same workbook, saved, and then renamed to Complex Conditions Example in the existing Conditions and Parameters Class Examples workbook.

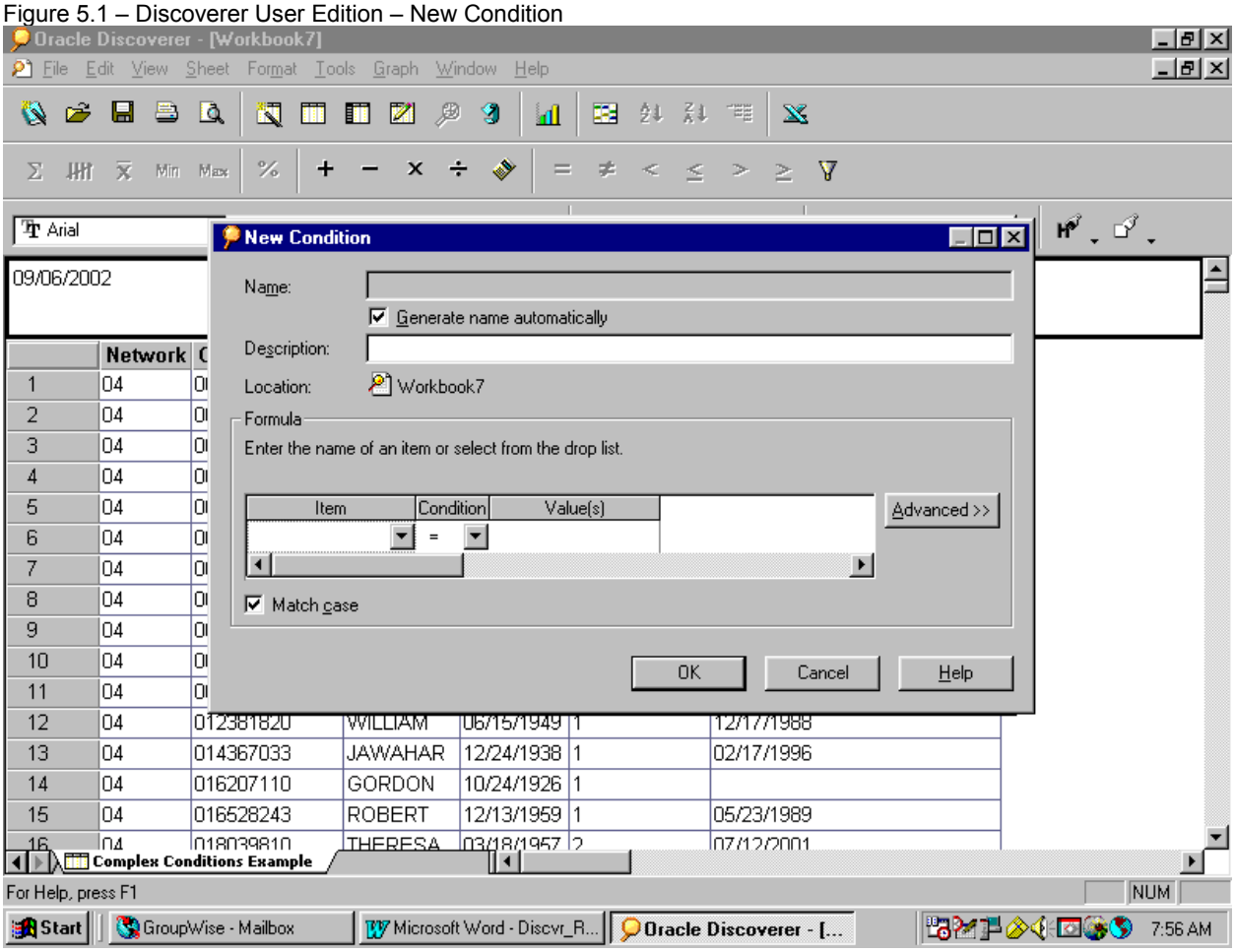
The complex condition we are going to create is one mentioned in the last topic. The two conditions we are going to combine are:

Network = ‘4’ OR
Network = ‘5’

Because coding this complex condition with an AND would cause the query to return no data (a patient cannot be in both Network 4 and Network 5), we will code this condition with an OR.

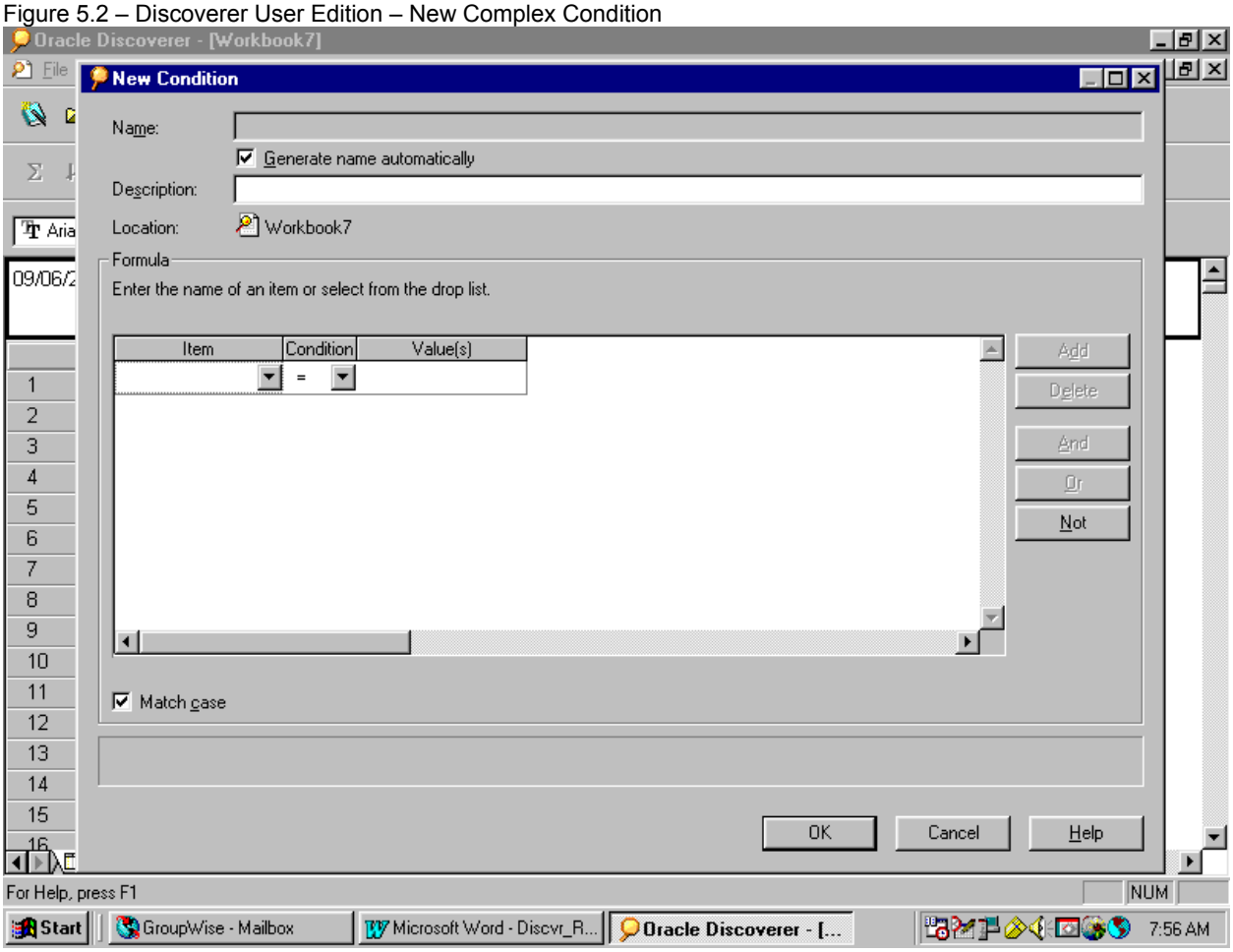
Complex conditions are created in much the same way as a simple condition. Open the Conditions window by selecting the Conditions option from the dropdown under the Tools menu item. To add a new condition, click the New… button to the right of the list of conditions. The New Condition window displays. (Figure 5.1)

Figure 5.1 – Discoverer User Edition – New Condition



Creating complex conditions begins with the same basic information that is entered for simple conditions. Enter the name and/or description, and indicate whether the name should be generated automatically and the case should be matched. To create the complex condition, click the Advanced >> button to open the complex condition entry window. (Figure 5.2)

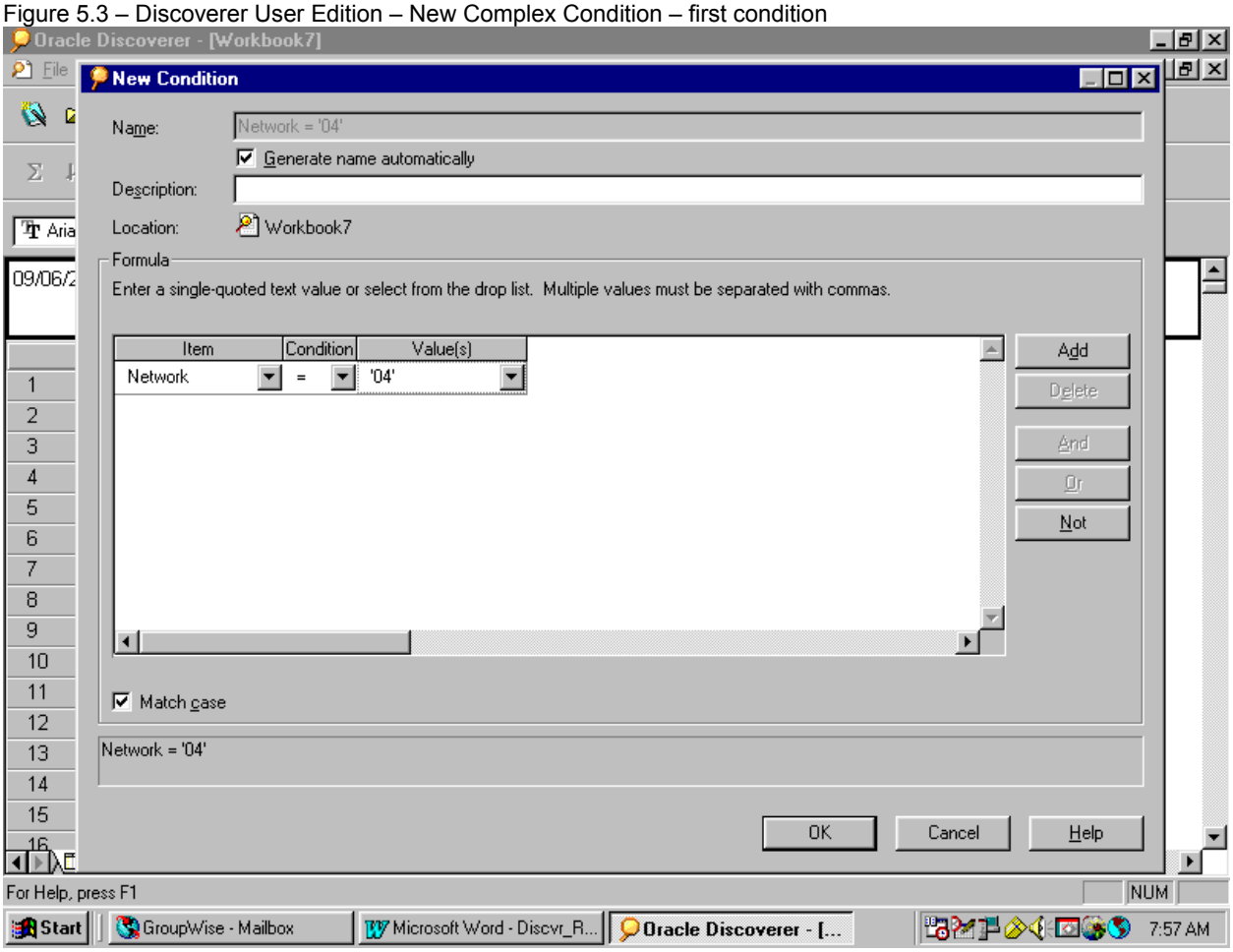
Figure 5.2 – Discoverer User Edition – New Complex Condition



Begin the definition of the complex condition by entering the Formula Item, Condition, and Value(s) for the first portion of the condition. In our example, we will select item 'Patient.Network', condition =, and value '04' for the first half of our complex condition.

Notice that when you complete at least one simple condition statement, the Add button on the right side of the formula box is enabled (Figure 5.3). Also enabled is the Not button. The Not button will reverse the Condition currently in effect. In this example, clicking the Not button would change the Condition from = to <>.

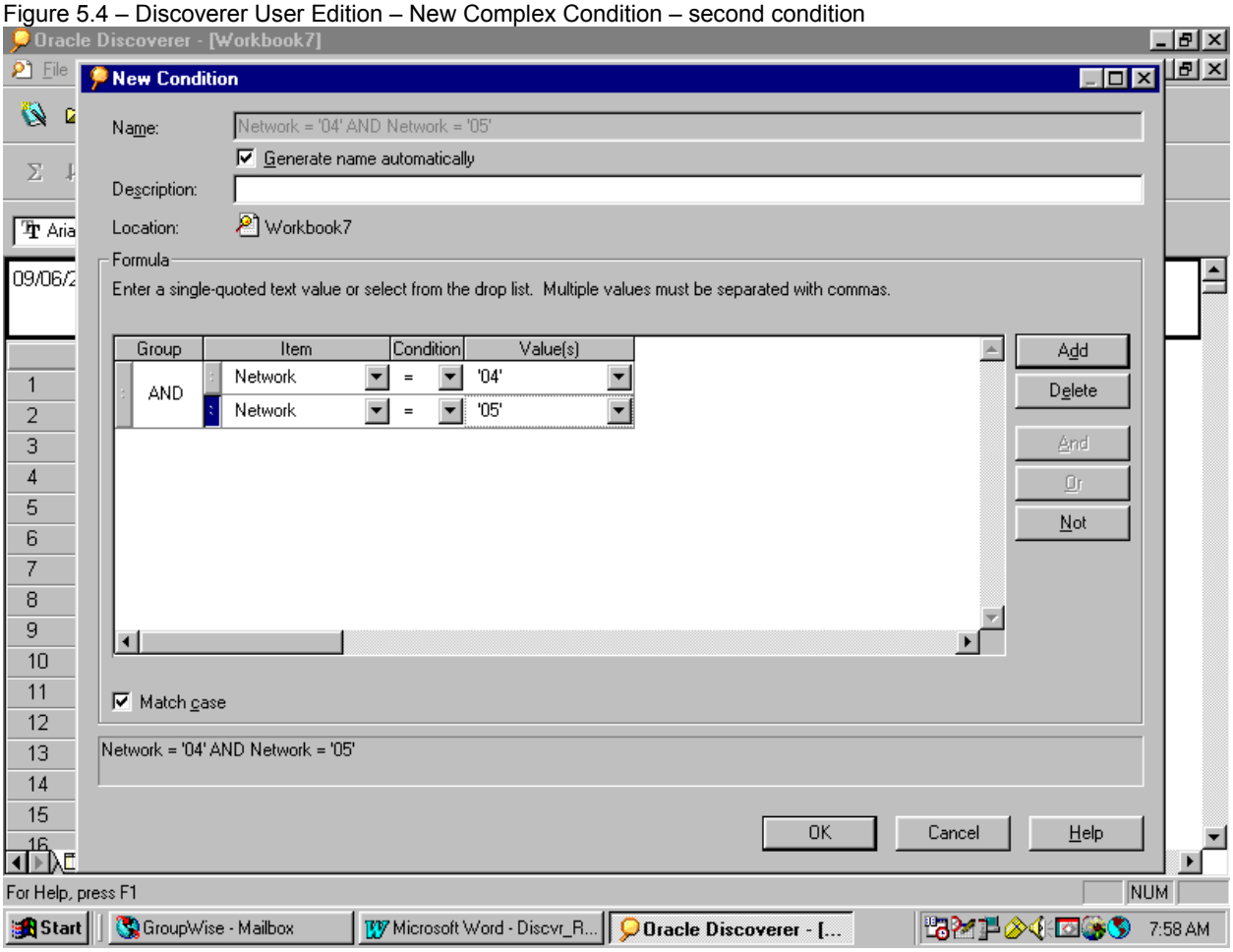
Figure 5.3 – Discoverer User Edition – New Complex Condition – first condition



To add the second half of the complex condition, click the enabled Add button on the right side of the Formula box. A second line appears, and you may continue to enter the rest of the complex condition. (Figure 5.4)

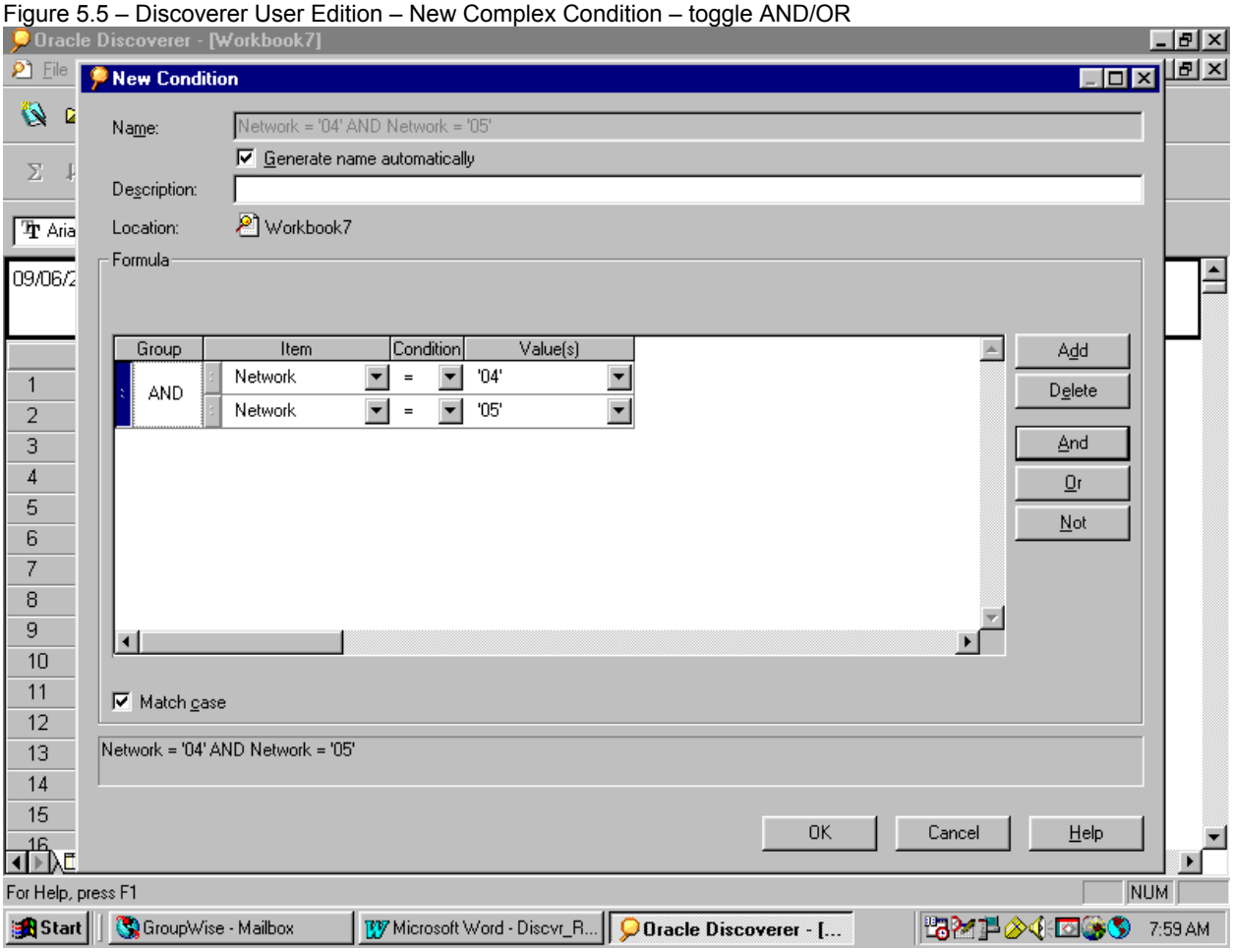
A Group column appears in the Formula box to indicate whether the simple conditions are to be compared in the query with an AND or an OR. By default, the two conditions are grouped with an AND.

Figure 5.4 – Discoverer User Edition – New Complex Condition – second condition



The Group value may be toggled between AND and OR in several ways. To enable the column, click the : row selection button to the left of the Group column value. (Figure 5.5) By activating the Group value, the And and Or buttons on the right side of the Formula box become enabled. You may toggle the Group values between AND and OR by clicking on the appropriate button, or by double clicking in the Group value area in the Formula box.

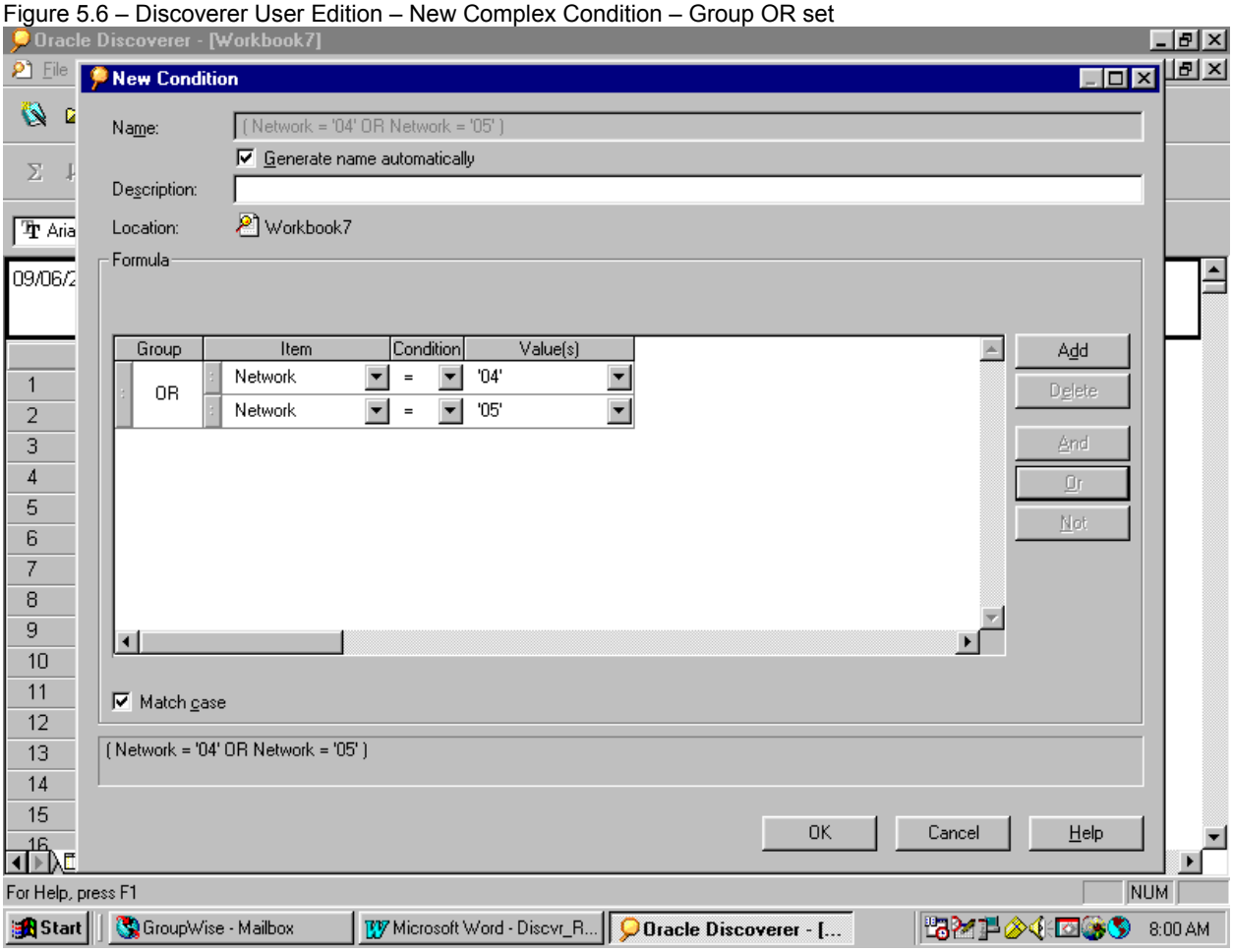
Figure 5.5 – Discoverer User Edition – New Complex Condition – toggle AND/OR



For this example, the Group value has been set to OR. (Figure 5.6)

Notice that because the Generate name automatically box is checked, Discoverer has automatically named this complex condition with the entire formula we've just defined. The Description has also defaulted to the entire formula as indicated in the box below the Formula box.

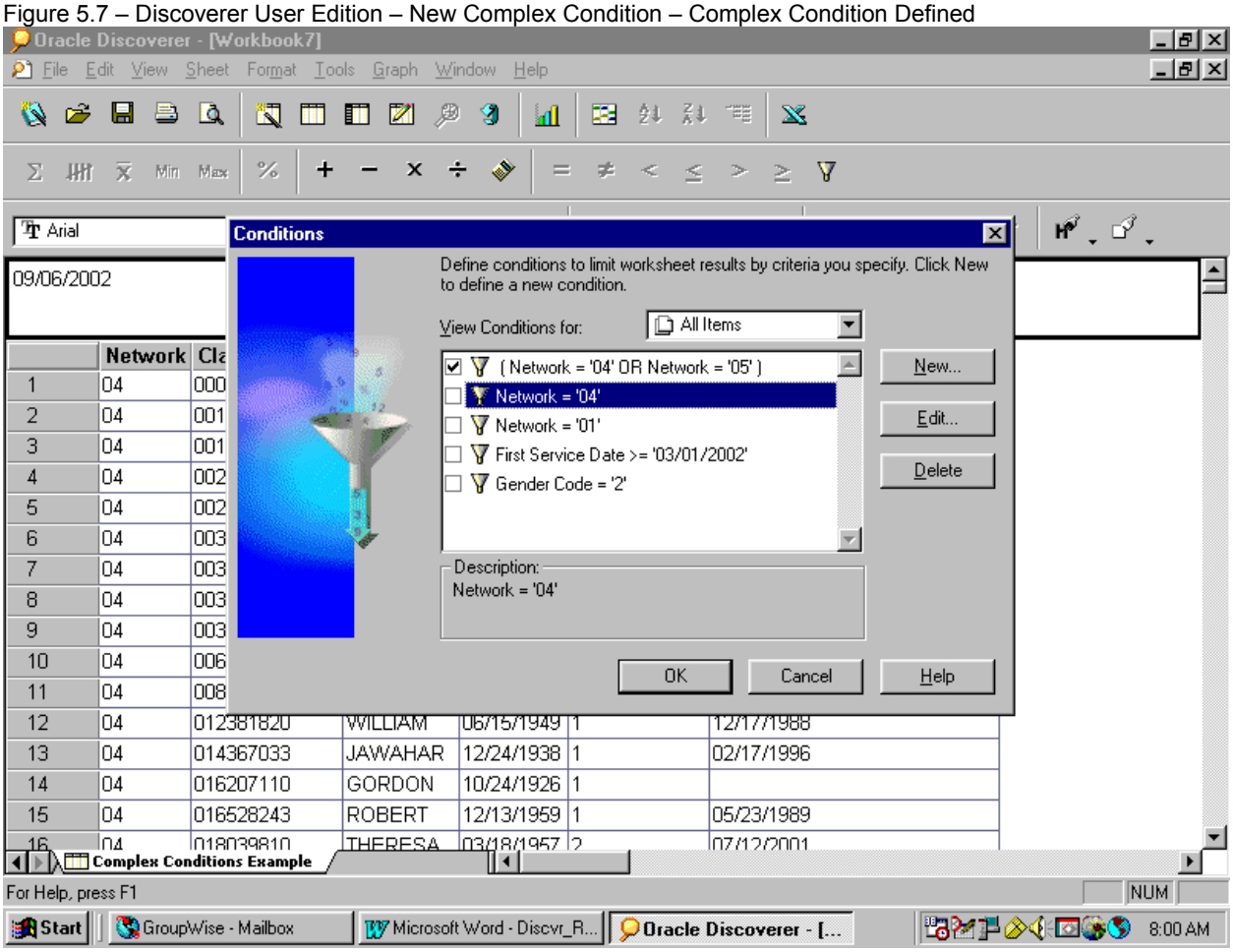
Figure 5.6 – Discoverer User Edition – New Complex Condition – Group OR set



Click OK to return to the Condition box, and notice that the complex condition has been added to the list and automatically activated for the query. (Figure 5.7) Because this worksheet was copied from the previous example, the State = 'DE' condition is also still activated. As you recall from the Select Predefined Conditions topic, click its check box to deactivate that condition.

Conditions in Discoverer workbooks are available for activation for all sheets in a workbook. This saves reentry time when adding additional sheets to an existing workbook where the filter requirements may be identical but the items to be displayed are different. Conditions defined in one workbook, however, are not visible to other workbooks.

Figure 5.7 – Discoverer User Edition – New Complex Condition – Complex Condition Defined



Lab Exercise 5

In this lab exercise, you will combine the conditions created in Lab 4 to build a basic complex condition.

1. Create a copy of the sheet you completed in Lab 4. Right click the Simple Conditions Lab 4 tab in your workbook, and select Duplicate as Table from the options list.
2. The Duplicate as Table window opens with current focus on the Table Layout tab page. At this point, you may make any modifications to layout, conditions, sort order, calculations, or even the selected items list. For this lab, make no changes yet and click OK to close this window.
3. Verify that the new sheet has been created, the query returns the same result set as the original sheet last returned, and the name of the sheet has defaulted to Simple Conditions Lab 4 2 (or possibly Simple Conditions Lab 5).

   (In Discoverer, every copy action you perform while building a query will cause the default object name to be identical to the original followed by an incremented sequence number. This keeps the objects (such as worksheet name within the workbook) unique in the metadata.)
4. Rename the new sheet to Complex Conditions Lab 5.
5. Open the Conditions window.
6. Click the New… button to the right of the conditions list box to create a new condition.
7. Optional: Enter a name and/or description for the new condition. If you do not enter anything in these fields, Discoverer will default their values to the Formula you enter.
8. Click the Advanced >> button to open the complex Condition edit window.
9. From the Item dropdown in the Formula box, select the Select Condition … option.
10. From the Conditions popup window, select Network = 04. Click OK to return to the complex Condition edit window.
11. Click Add to add another condition line to the Formula box.
12. From the Item dropdown in the new line, select the Select Condition … option.
13. From the Conditions popup window, select Transplant Dates in 2002. Click OK to return to the complex Condition edit window.
14. Click OK to return to the main Conditions window.
15. Verify that the new complex condition has been created and is activated for the query. Deactivate the single conditions still in effect from the original worksheet.
16. Click OK to return to the worksheet and execute the query.
17. How many rows returned?
18. Open the Conditions window again. Click the name of the complex query you just created to highlight it. Open it in the Edit Conditions window by clicking the Edit button to the right of the condition list.
19. Click in the Group column so that the : button to the right of the group value is highlighted. Change the AND to OR by double clicking on the AND group value, or by clicking the OR button on the right side of the Formula box. Click OK to return to the main Conditions window.

20. Click OK to return to the worksheet and execute the query.
21. Now how many rows are returned? Why?
22. Save your workbook in its current state.

Important note: When you create a complex condition using previously defined simple conditions as we did in this example, the existence of the complex condition relies solely on the existence of each of the simple conditions it contains. Be very careful when deleting simple conditions from workbooks. If you delete a simple condition that has been used in a complex condition, not only will the simple condition disappear from the workbook, it will take the complex condition along with it!

Topic 6 – Grouping Complex Conditions

Sometimes the statement of business need is complicated enough that a single-level complex condition is not enough. Remember the earlier example where you wanted to create a REMIS report showing selected patients in Network 1 whose most recent transplant was on or after 01/01/2002 in the same report with selected patients in Network 2 whose most recent transplant was on or after 03/01/2002? The filters required for this business need are multi-level:

```
            Network = '1' AND
            Most Recent Transplant Date >= '01/01/2002'
OR
            Network = '2' AND
            Most Recent Transplant Date >= '03/01/2002'
```

Multi-level filters are defined using grouped complex conditions. An easy way to mentally picture a grouped complex condition is to use the example of a mathematical expression. A very complicated mathematical expression to be more precise. In complex expressions, lower level components are identified with the order in which they are to be computed by their own set of parentheses. For example, in the expression:

```
( (a + b) – (c * ( (d + e + f) – (g * h) ) ) ) / i
                   (   #1    ) -  (  #2 )
             c *   (           #3        )
(  #5   ) -  (           #4              )
(                 #6                          )/ i
```

the parentheses help you to identify the numerous lower level components that must be solved prior to solving the higher level components. In the example above, both the product #2 of (g * h) and the sum #1 of (d + e + f) must be computed before the product may be subtracted from the sum leaving the difference #3. Next, the product #4 of (c * #3) is determined along with the sum #5 of (a + b). Finally, the difference #6 of (#5 - #4) can be divided by i.

In our expression example, without the parentheses, the outcome might be very different depending on who was solving it. The same is true for setting up grouped complex conditions. Unless you get them set up appropriately within the levels, the query result set may produce very different results. Computer programs handle mathematical functions according to default rules dictating the order of operations in the absence of parentheses. Likewise, Oracle database queries default the order in which it evaluates each distinct condition in the WHERE clause in the absence of parentheses. It resolves AND conditions first, followed by OR conditions. To remove any possibility of doubt, particularly for very complicated grouped complex conditions, be very specific – and very careful - in your definition.

Back to our multi-level filters example. The statement of business need has already been analyzed, and the filters required are:

Network = ‘01’ AND
Most Recent Transplant Date >= ‘01/01/2002’
OR
Network = ‘02’ AND
Most Recent Transplant Date >= ‘03/01/2002’

After defining this complex condition in Discoverer, it will be displayed in the default condition definition/name as the following (although not lined up quite like this):

```
( ( (Network = ‘01’) and
    (Most Recent Transplant Date >= '01/01/2002') ) or
  ( (Network = ‘02’) and
    (Most Recent Transplant Date >= ’03/01/2002') ) )
```

Discoverer makes use of parentheses to accurately relay intent to Oracle queries just as mathematicians use parentheses to accurately relay order of solution to mathematical expressions. The advantage you have with Discoverer is that you don’t have to code them. Discoverer automatically embeds the parentheses into the query filters you create through the complex condition wizards, behind the scene, creating the WHERE clause for the SQL SELECT statement that actually retrieves the data. The trick for you is in getting the complex condition levels created appropriately to relay your true intent to Discoverer.

To begin the process of defining a grouped complex condition, parse the filter requirements in your statement of business need visually. The method you use to picture the filter doesn't matter – use whatever method you are comfortable with. If you are mathematically inclined and prefer expressions and parentheses over hierarchical lists, use them. For this example, we will use the hierarchical list:

Network = '01' AND
Most Recent Transplant Date >= '01/01/2002'
OR
Network = '02' AND
Most Recent Transplant Date >= '03/01/2002'

For our class example, we have copied the Complex Conditions Example to another worksheet in the Conditions and Parameters Class Examples Workbook, and renamed it to Grouped Complex Conditions Example. Next, we opened the Conditions window, and clicked New to create a new condition.

Creating grouped complex conditions begins like all other conditions with the entry of the basic name and/or description text, automatic name generation, and match case settings. It is suggested that for grouped complex conditions, the description be allowed to default to the actual formula expression. This will be handy in debugging any problems you may encounter with a grouped complex condition.

Open the edit complex condition window by clicking the Advanced >> button.

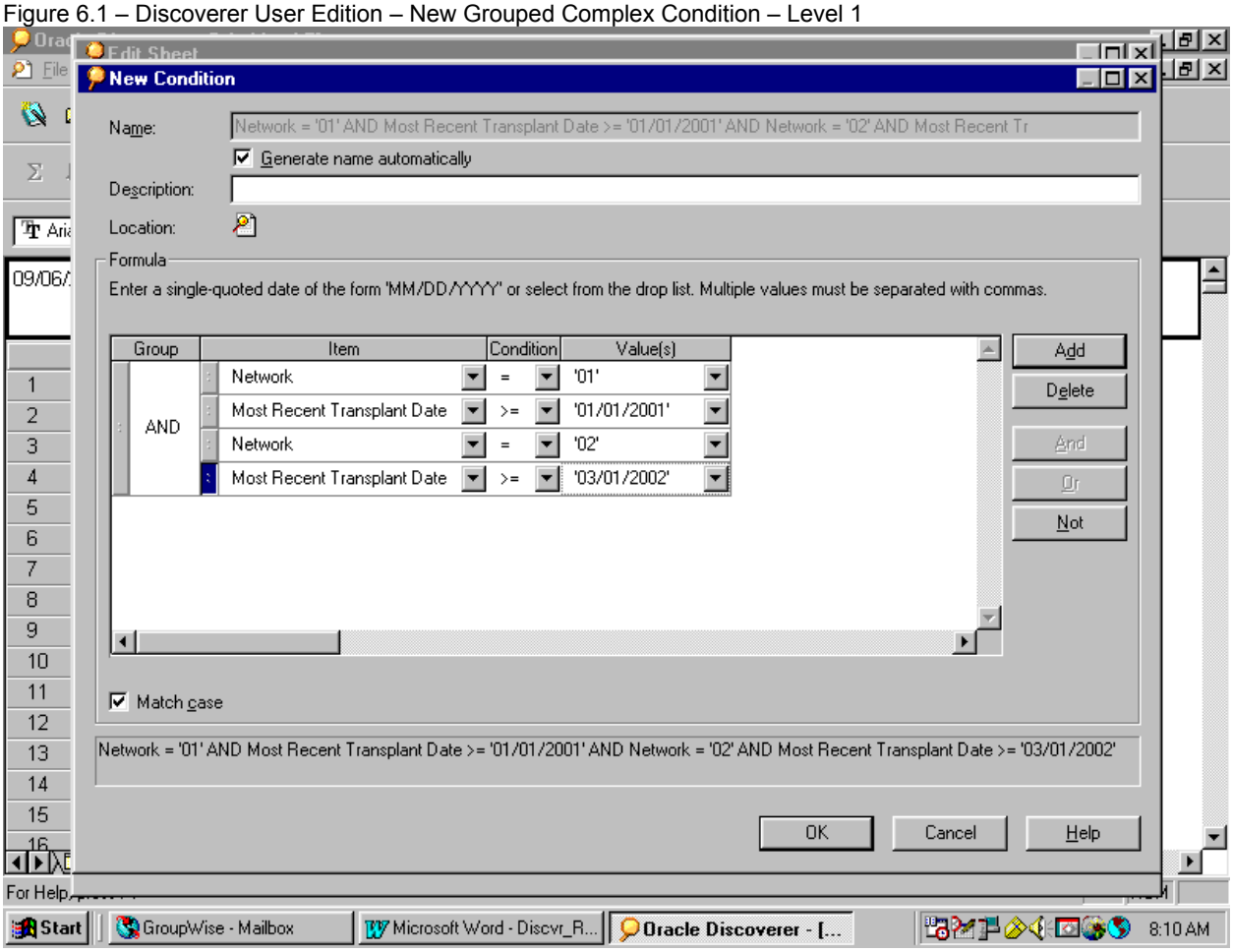
Examine the grouped complex condition diagram you have developed, and decide how you want to proceed with entering the individual conditions. This is entirely a matter of personal preference. There are probably as many methods for entering grouped complex conditions as there are Discoverer users. For Discoverer beginners, however, it is recommended that you enter all of the simple conditions first, allowing them to group together into one with the default AND group value. After all of the simple conditions have been entered, add the sublevels to the grouped complex condition one at a time, working from the bottom up and outside in.

In our example, we are only dealing with two Group levels:

Level 2	Level 1	
		Network = '01'
	AND	Most Recent Transplant Date >= '01/01/2001'
OR		
		Network = '02'
	AND	Most Recent Transplant Date >= '03/01/2002'

Entry will begin with adding all of the conditions for Level 1 first. (Figure 6.1)

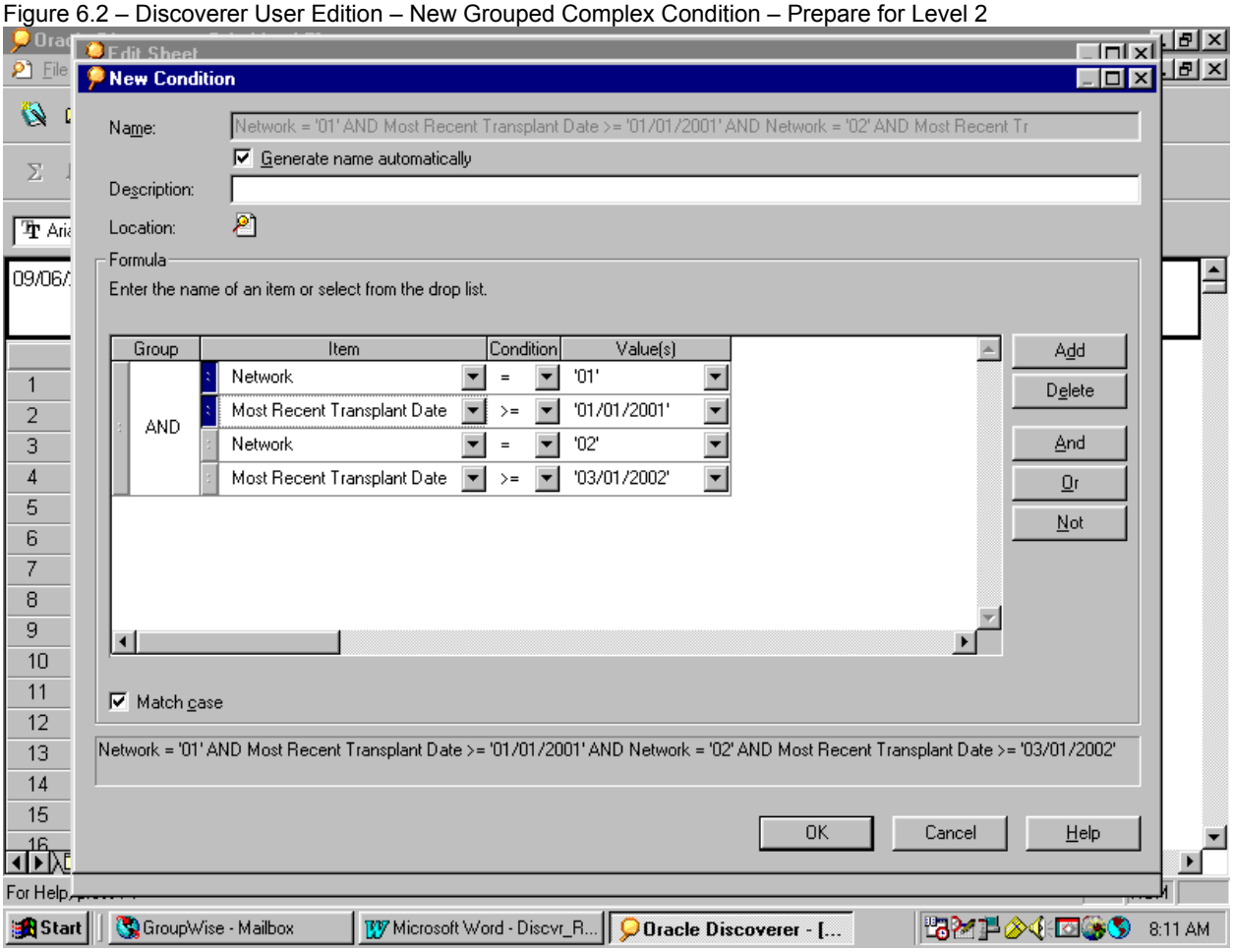
Figure 6.1 – Discoverer User Edition – New Grouped Complex Condition – Level 1



Note that the formula has been created without parentheses. This is because all of the conditions are still at the same level.

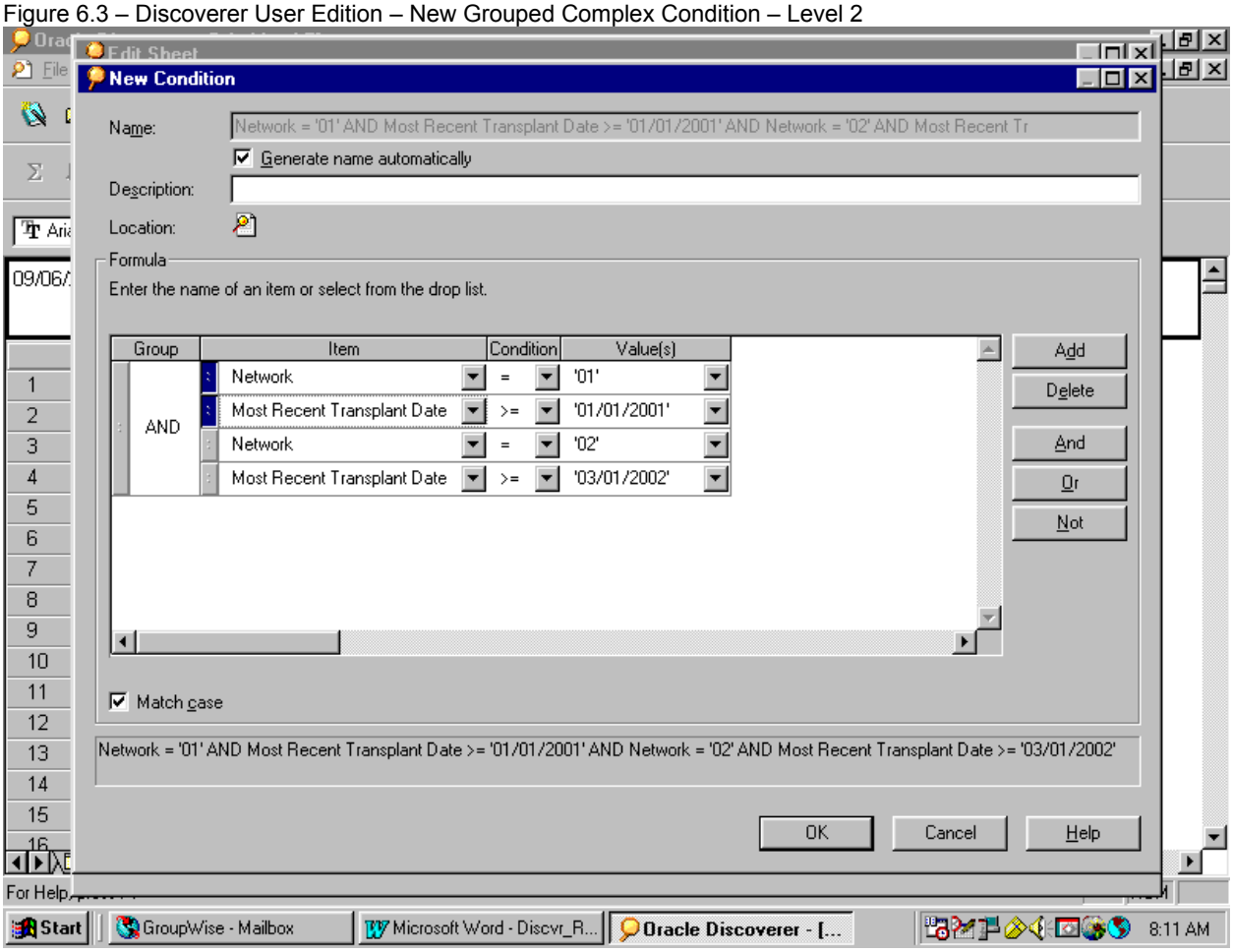
To add the second level group value in Discoverer, highlight the first set of two or more conditions to be grouped together by holding down the Ctrl key and clicking the : row selection button directly to the left of each of the Item columns to be grouped. (Figure 6.2)

Figure 6.2 – Discoverer User Edition – New Grouped Complex Condition – Prepare for Level 2



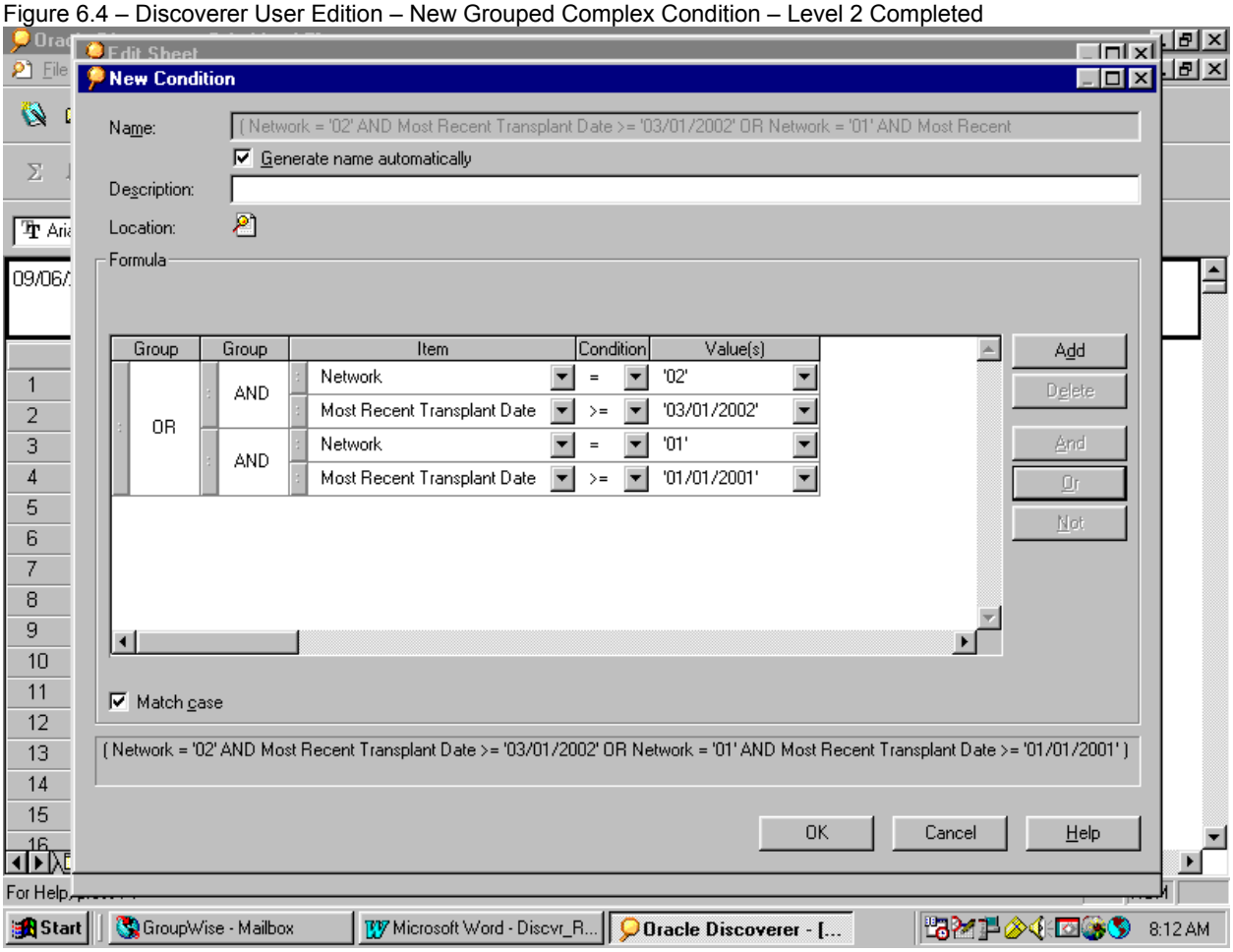
With the two or more conditions highlighted, group them in the same level by clicking the And button. (Figure 6.3)

Figure 6.3 – Discoverer User Edition – New Grouped Complex Condition – Level 2



Repeat this process for the remaining conditions/groups required for your filter. As appropriate, toggle the Group values for each level according to your filter requirements. For our example, we left the lowest level group items with the AND group value, and coded the higher-level group value from AND to OR. (Figure 6.4)

Figure 6.4 – Discoverer User Edition – New Grouped Complex Condition – Level 2 Completed



Note that the Description for the formula now contains parentheses. Because this is a simplified version of a grouped complex condition, and Discoverer knows that by default Oracle evaluates ANDs before ORs in a WHERE clause, it has only coded this condition with one set of parentheses.

Lab Exercise 6 – Creating Grouped Complex Conditions

For this lab exercise, we are going to use the following statement of business need and filter requirements to build a grouped complex condition.

Statement of Business Need:

I need a list of all patients in Network 6 that have received a transplant since 02/01/2002 from a cadaver donor. In the same report, I also want a list of all patients in Network 7 that have received a transplant since 03/01/2002 from a living donor. For each patient, I want to see their claim number, name, state, Network, transplant date, and donor type.

Filter Requirements Diagram:

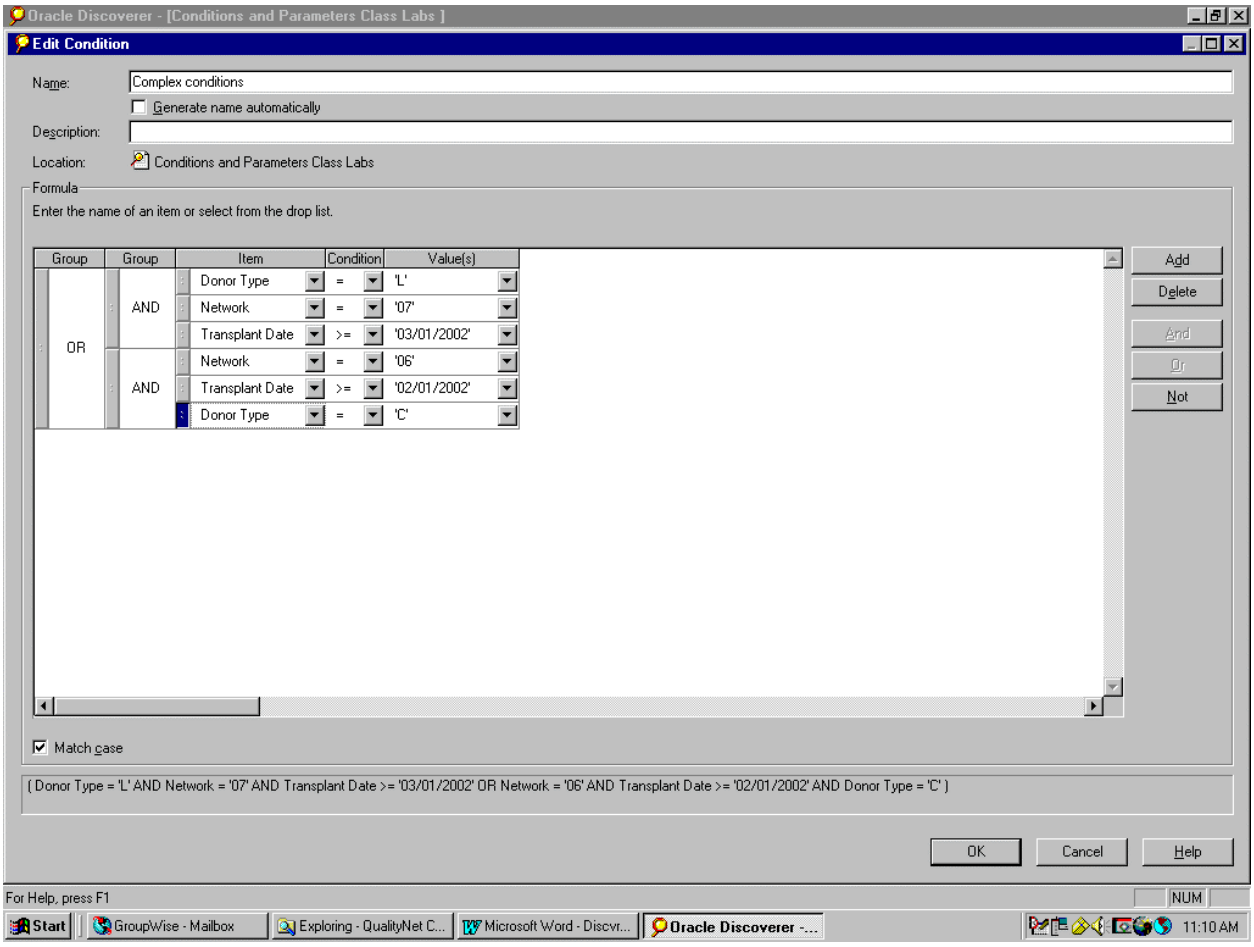
Level 3	Level 2	Level 1	Condition
			Network = ‘06’
	AND		Transplant Date >= '02/01/2002'
		AND	Donor Type = ‘Cadaver'
OR			
			Network = ‘07’
	AND		Transplant Date >= '03/01/2002'
		AND	Donor Type = 'Living'

1. Create another copy of the sheet you completed in Lab 4. When the Duplicate as Table window opens, make no changes to the sheet and click OK to close the window.
2. Rename the new sheet to Grouped Complex Conditions Lab 6.
3. Open the Conditions window.
4. Click the New… button to create a new condition.
5. Optional: Enter a name for the new grouped complex condition. Leave the Description blank so that you can examine the resulting condition text created by Discoverer.
6. Open the complex Condition edit window. (Hint. Click the Advanced >> button)
7. Add the simple conditions as follows:

Item	Condition	Value(s)
Network	=	‘06’
Transplant Date	>=	’02/01/2002’
Donor Type	=	’Cadaver'
Network	=	‘7’
Transplant Date	>=	’03/01/2002’
Donor Type	=	’Living’

8. Add the first group to the condition. Beginner Discoverer users: build the groups for the grouped complex condition from the bottom up, and outside in. Holding down the Ctrl key, click on the : row selection button to the left of each of Network = '06', Transplant Date >= '02/01/2002', and Donor Type = 'Cadaver'.
9. Click the AND button to the right of the Formula box. Note that the rows move to the top of the Formula box and that a new Group level with a value of AND is added.
10. Add the second group to the condition. Holding down the Ctrl key, click on the : row selection button to the left of each of Network = '07', Transplant Date >= '03/01/2002', and Donor Type = 'Living'.
11. Click the AND button to the right of the Formula box. Note that the rows move to the top of the Formula box and that a new Group level with a value of AND is added.
12. Add the third and final group to the condition to place the either/or option between the two groups you just created. Holding down the Ctrl key, click on the : row selection button to the left of each of the AND group values related to the combination Network/Transplant Date/Donor Type conditions.
13. Click the OR button to the right of the Formula box. Note that the rows do not change, but that a new Group level combining the other two groups with a value of OR is added.
14. Look at the complete Formula as it is displayed in the Description box at the bottom of the edit window. How many sets of parentheses does it contain? Does this formula reflect the filters identified from the original statement of business need?
15. Click OK to return to the Conditions window.
16. Verify that the new grouped complex condition has been created and is activated for the query. Deactivate any other conditions still in effect from the original worksheet.
17. Click OK to return to the worksheet and execute the query.
18. How many rows were returned? Did they fit the filters identified from the original statement of business need?

Lab Exercise 6 Solution



Topic 7 – Create Parameters

As we have learned in the previous topics, conditions are a valuable tool to use to limit the amount of data returned from a query in Discoverer. One of the limitations of conditions, however, is that they are coded with a specific value to be used as a filter. Each time you want to run the query using that condition but with a different value, you have to edit the condition and change that value. If a query only uses one condition, that may not be a bad solution. However, if the query uses multiple conditions for which you need to change the values, the task becomes very cumbersome. Parameterized conditions solve this problem.

Parameterized conditions provide the ability to set condition values at run time, without having to edit the conditions themselves. To enable a condition with parameters, the Value(s) normally entered in the condition definition are replaced with a Parameter. The parameter is presented to the user at run time via a response window. If there is more than one parameter coded for a worksheet, Discoverer groups all of them in the same response window, and requires at least one entry into each setting prior to leaving the window. Parameters may be defined linking the value to an item in a folder in use in the worksheet, or they may be defined without a link, allowing them to be passed as arguments into database functions that are part of the query. Parameters may be set up with the number of allowed Value(s) limited to only one, or they may be set up allowing multiple Value(s) to be selected.

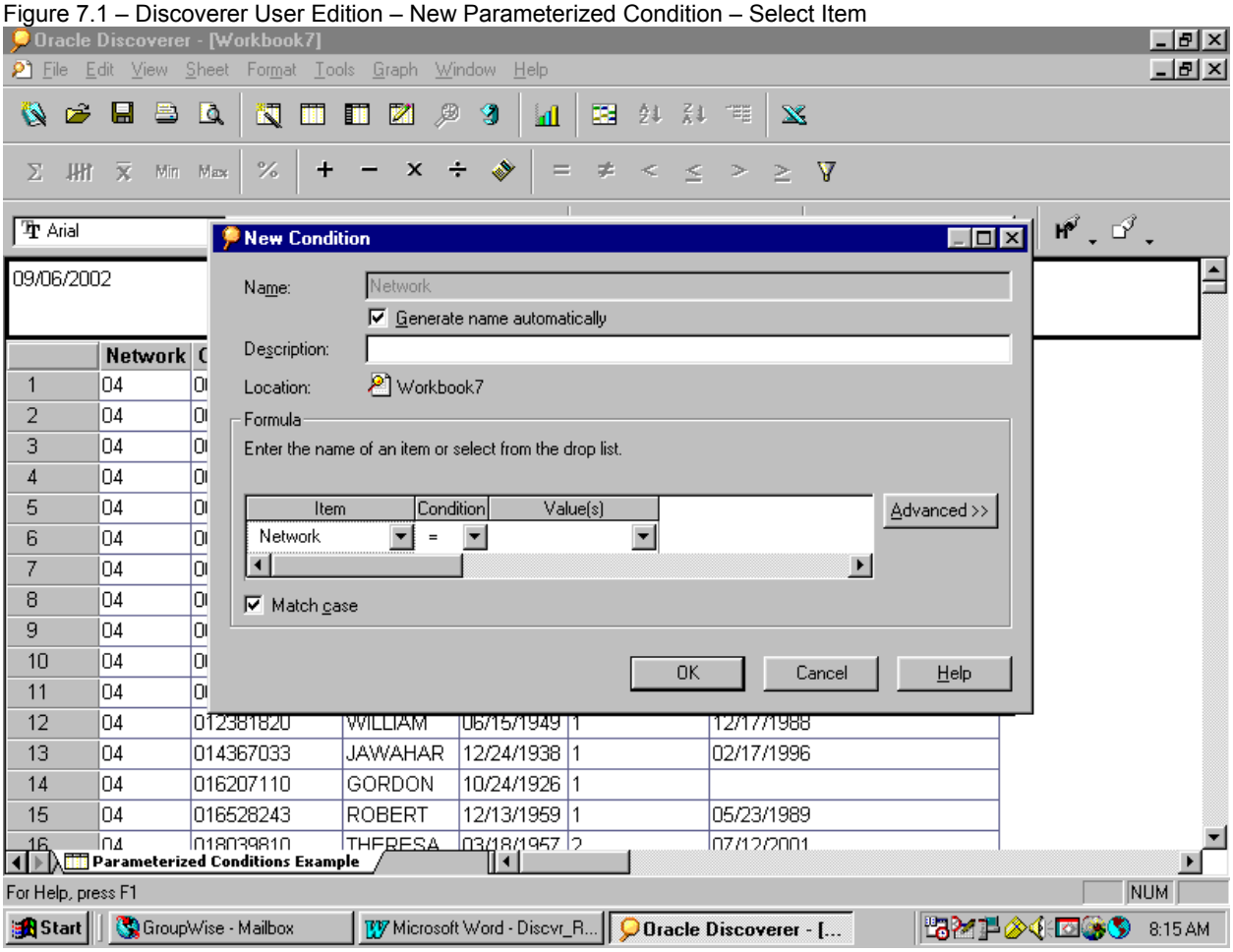
For our class example on Parameters, we have copied the Simple Conditions Example to another worksheet in the Conditions and Parameters Class Examples Workbook, and renamed it to Parameterized Conditions Example. Next, we opened the Conditions window, and clicked New to create a new condition. Although it is possible to change an existing simple condition into a parameterized condition, for demonstration purposes in this class, we will create a new one.

Creating parameterized conditions begins like all other conditions with the entry of the basic name and/or description text, automatic name generation, and match case settings.

In this example, we're going to use the premise that each time this sheet is to run, we want to be able to change the Network value(s) that are being reported. Instead of the current condition of selecting only Network 4, we also want to be able to query for any of the other Networks.

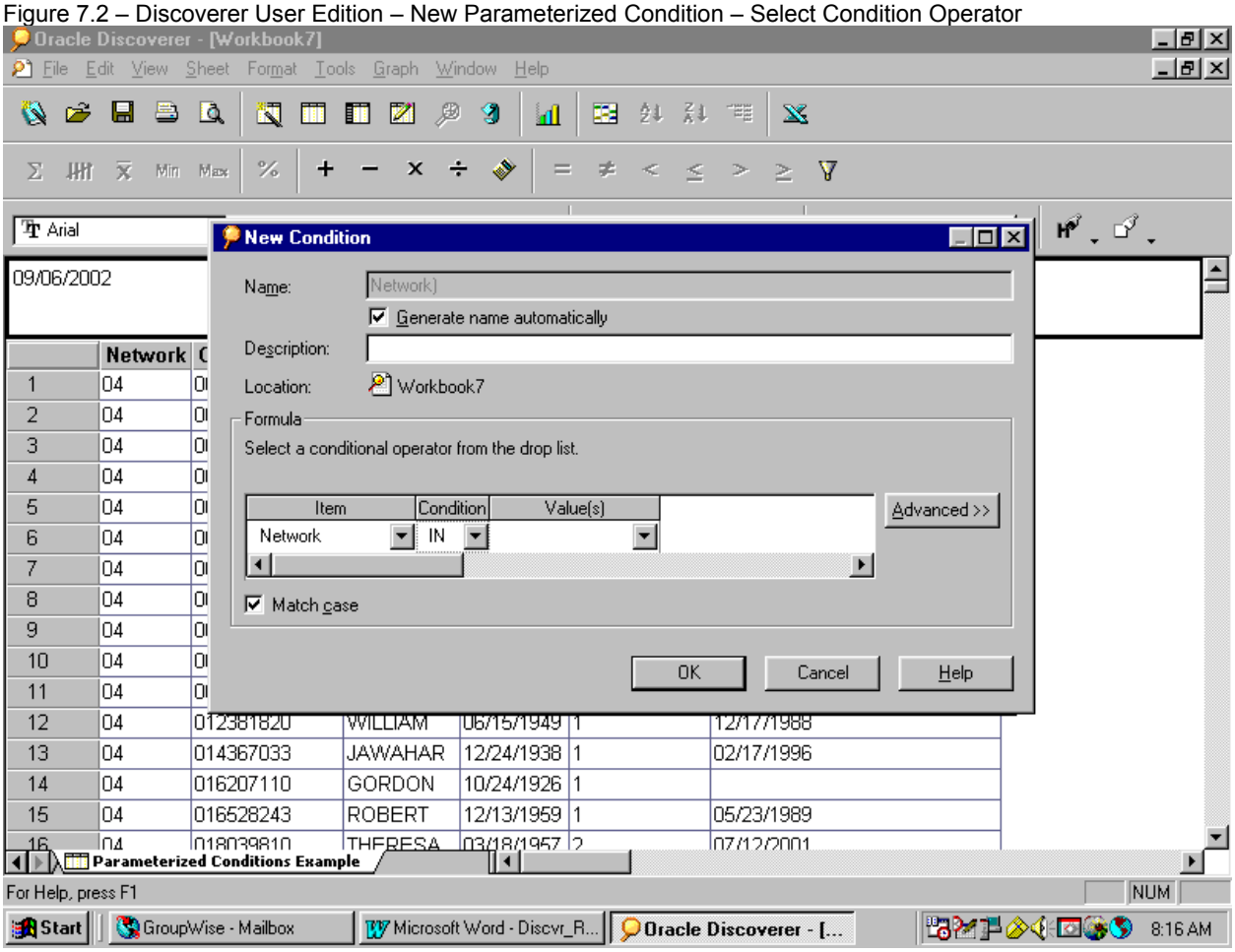
To create a parameterized condition, first identify the Item to be associated with the parameter. In this example, we have selected the Network item. (Figure 7.1)

Figure 7.1 – Discoverer User Edition – New Parameterized Condition – Select Item



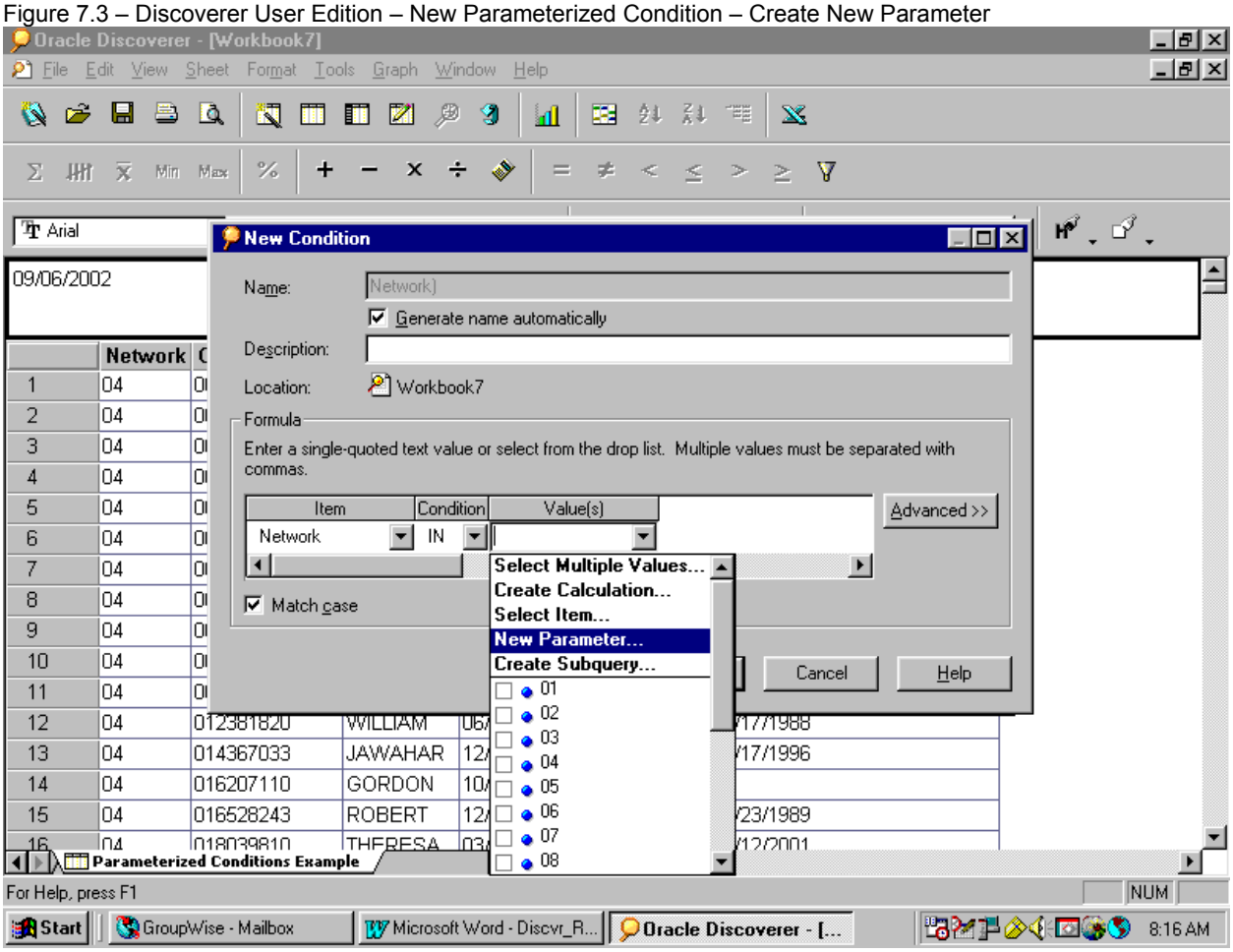
Next, choose the Condition operator. The default is =. Determine whether the parameter will allow only a single or multiple Value(s) to be evaluated in this condition. If you will be allowing multiple values and comparing them for equivalency, use the IN operator rather than the =. (Figure 7.2)

Figure 7.2 – Discoverer User Edition – New Parameterized Condition – Select Condition Operator



Finally, associate the Value(s) with a new parameter. Open the list under the Value(s) heading, and select New Parameter from the options. (Figure 7.3)

Figure 7.3 – Discoverer User Edition – New Parameterized Condition – Create New Parameter



The new Parameter window opens. There are several entries to be made on this window in order to make the parameter user-friendly at run time. First, note that the Condition Item and Operator are displayed on this window for your reference, but have been protected from entry. The Name field provides the opportunity to name the parameter. The Prompt field is where you enter the text label that is displayed to the left of the parameter field at run time in the Parameter Wizard. Make this label as succinct as possible – there is not a lot of screen real estate in the Parameter Wizard. The Description field is where additional information may be entered related to the parameter. This information displays at the bottom of the Parameter Wizard box at run time for each parameter item as it comes into focus. The Multiple Values option is checked enabled by default. Finally, you may optionally select a default value, or values if you checked the multiple values option box, for the parameter. The dropdown list contains the same set of values based on the Item selected for the condition as you would find in the Condition edit window. Whether or not you select default values at this time, remember this: the values you enter the last time you run the query prior to saving it will be stored as the default values for the next time the worksheet is run. (Figure 7.4)

Figure 7.4 – Discoverer User Edition – New Parameterized Condition – New Parameter Dialog

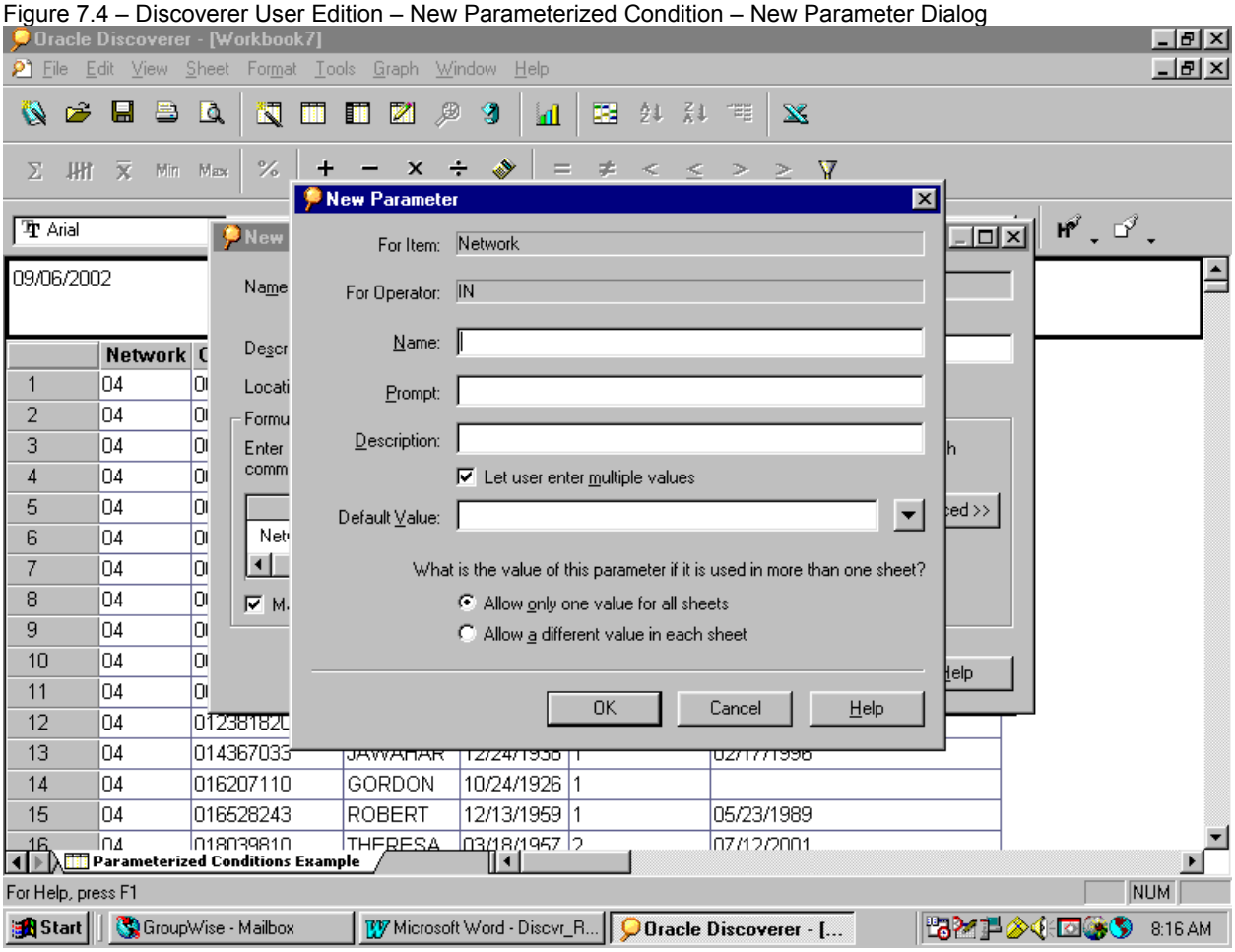
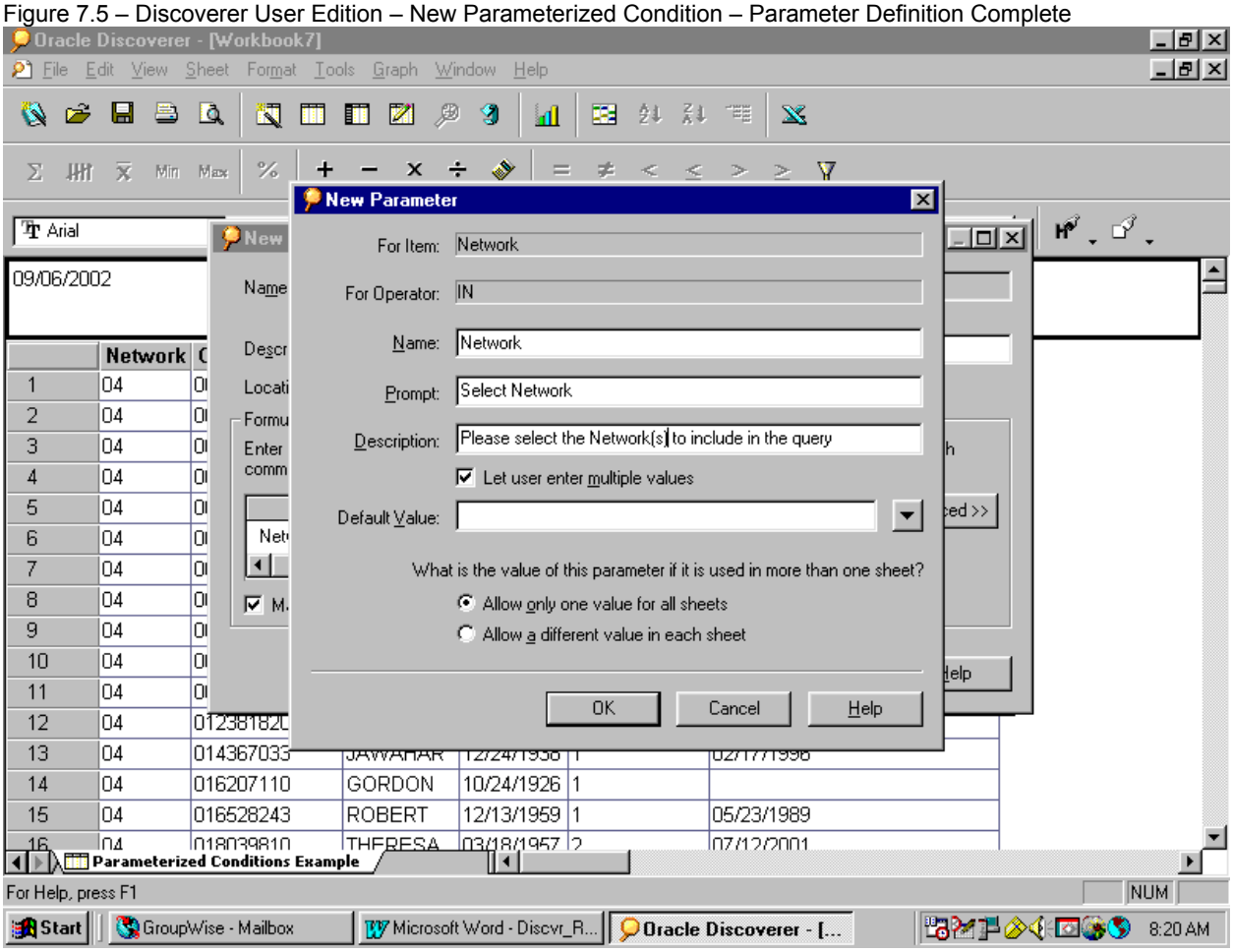


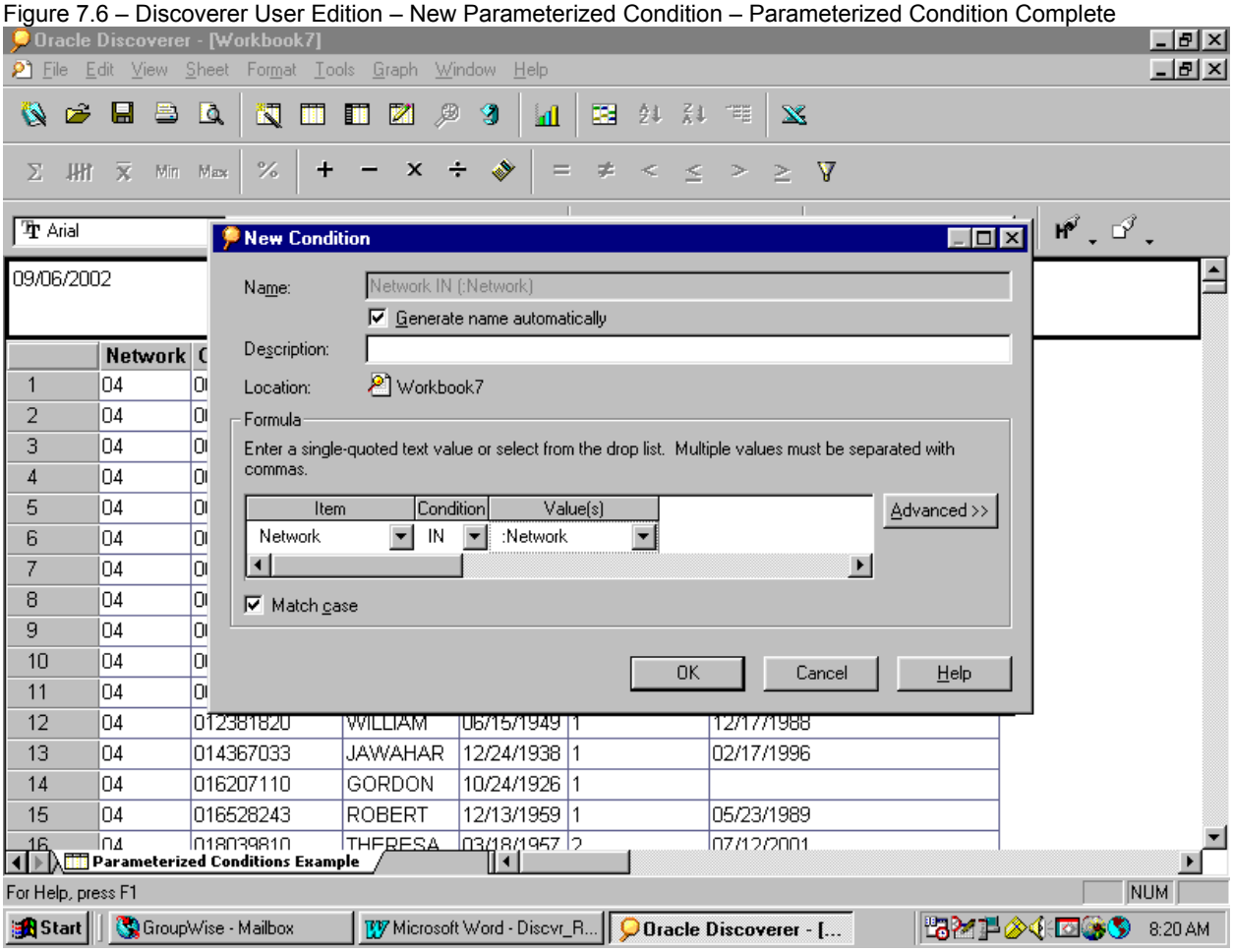
Figure 7.5 shows the New Parameter entries as they have been coded for the example. Click OK to return to the Condition edit window.

Figure 7.5 – Discoverer User Edition – New Parameterized Condition – Parameter Definition Complete



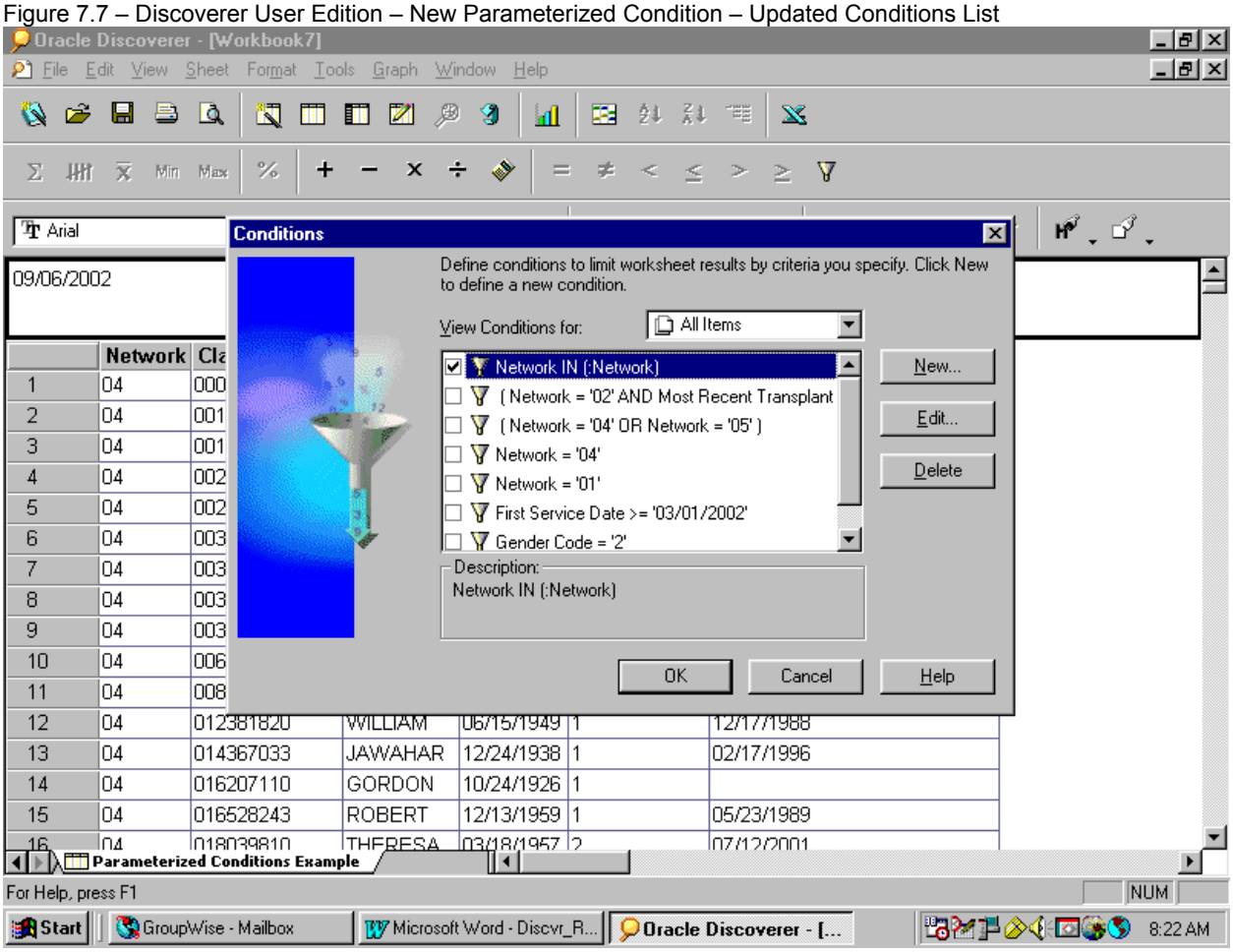
The New Condition edit window displays again, showing the newly created parameter in the Value(s) column for the condition. Note the difference in format over what would be shown if actual values were coded. Parameters are prefixed with a : followed by the parameter name entered on the Parameters edit window. (Figure 7.6) Click OK to return to the main Condition window.

Figure 7.6 – Discoverer User Edition – New Parameterized Condition – Parameterized Condition Complete



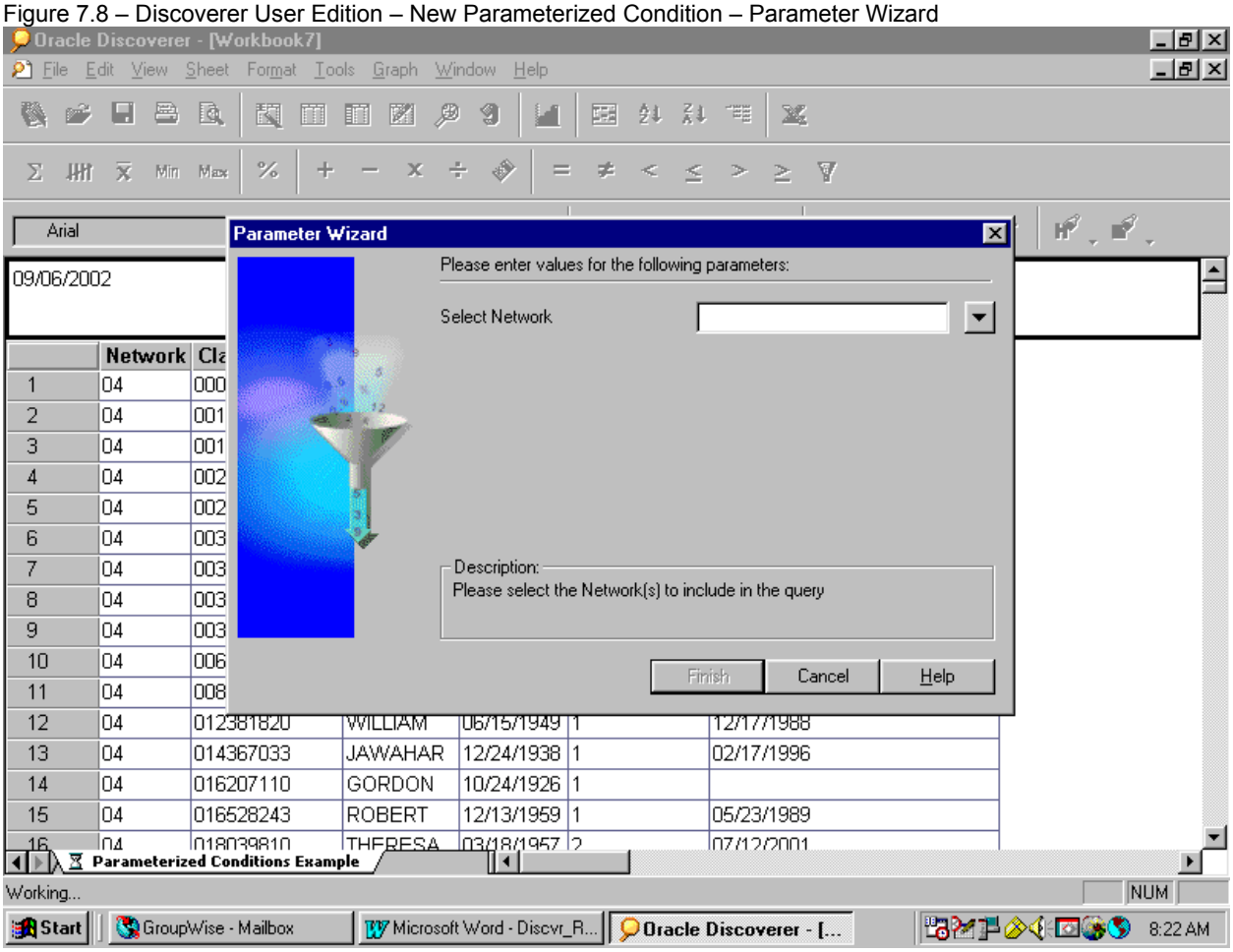
The main Conditions window is displayed. At this time, the new parameterized condition has been saved and activated for the sheet. The previous condition in effect for the original Simple Conditions Example worksheet, Network = '4', has been disabled. Click OK to return to the worksheet. (Figure 7.7)

Figure 7.7 – Discoverer User Edition – New Parameterized Condition – Updated Conditions List



Now when the sheet runs, the Parameter Wizard box appears first. Note that until entries are made in the parameters on the page, the Finish button is disabled. Parameters are required fields to be entered before a worksheet will run the query. (Figures 7.8)

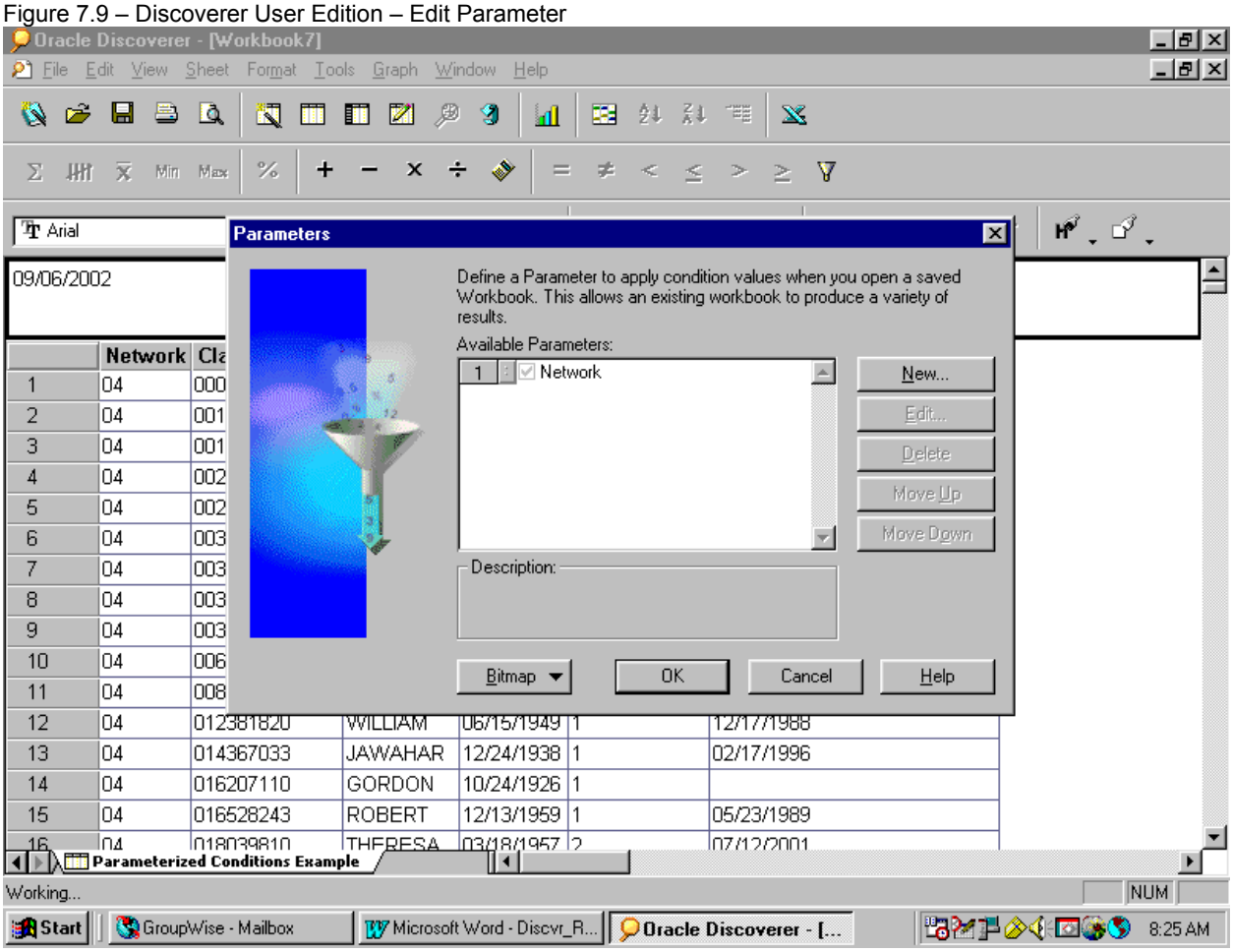
Figure 7.8 – Discoverer User Edition – New Parameterized Condition – Parameter Wizard



There are actually multiple methods that may be used to create parameters in Discoverer. One method in addition to the Conditions edit just demonstrated is the Parameters tool, which may be accessed via the Tools menu. The Parameters window lists all of the parameters that have been defined for the workbook. It lists the parameters in the workbook by name, along with a check box indicating which of the parameters are active for the current sheet. The checkbox may be protected as it is in this example, indicating that the parameter is in use by a condition currently active for the worksheet.

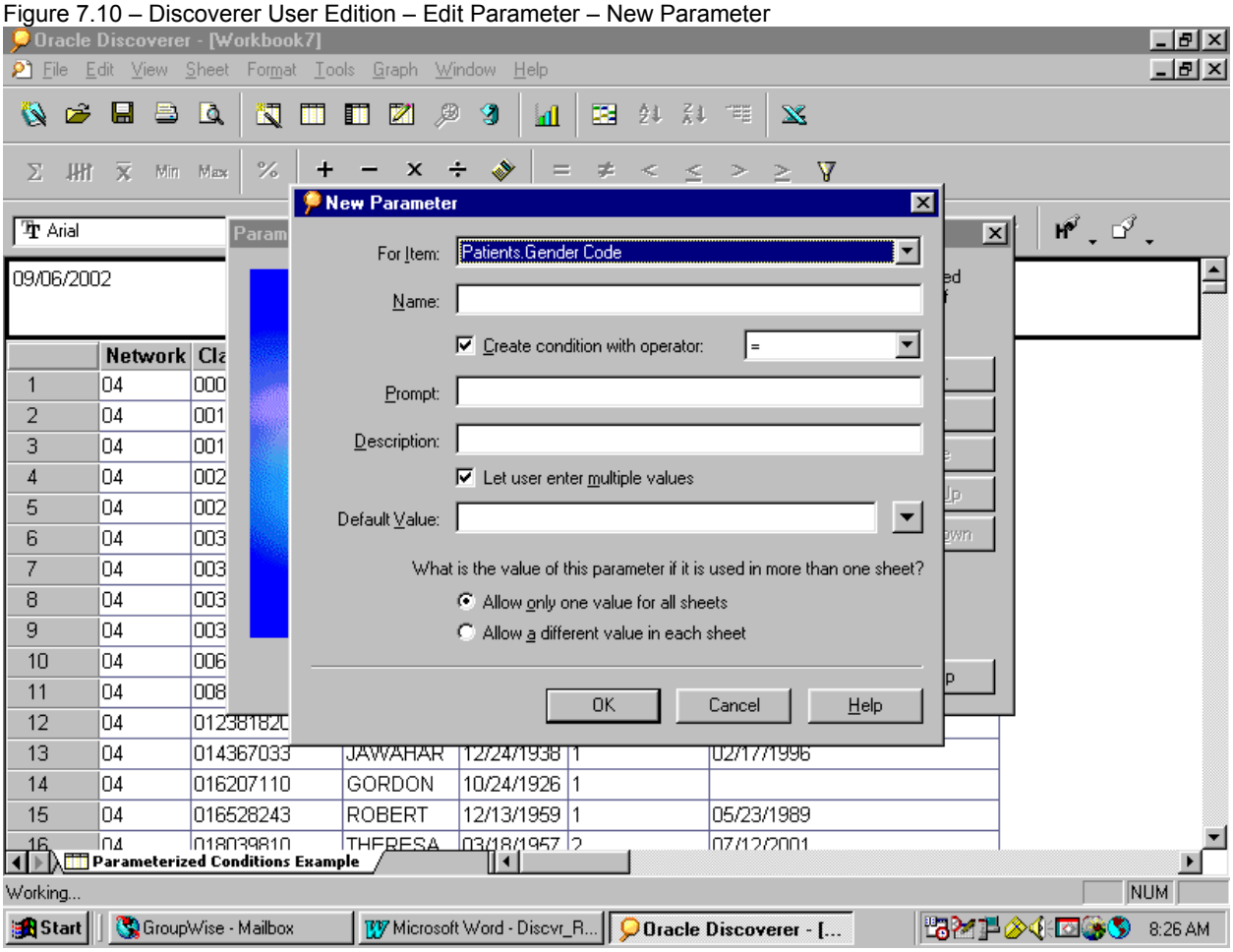
From this window, a parameter may be added, edited, or deleted. If there are multiple parameters defined for a worksheet, the order in which they appear in the Parameter Wizard for the end user is set here using the Move Up and Move Down buttons. Since there is only one parameter available at this point in the example, those buttons are disabled. (Figure 7.9)

Figure 7.9 – Discoverer User Edition – Edit Parameter



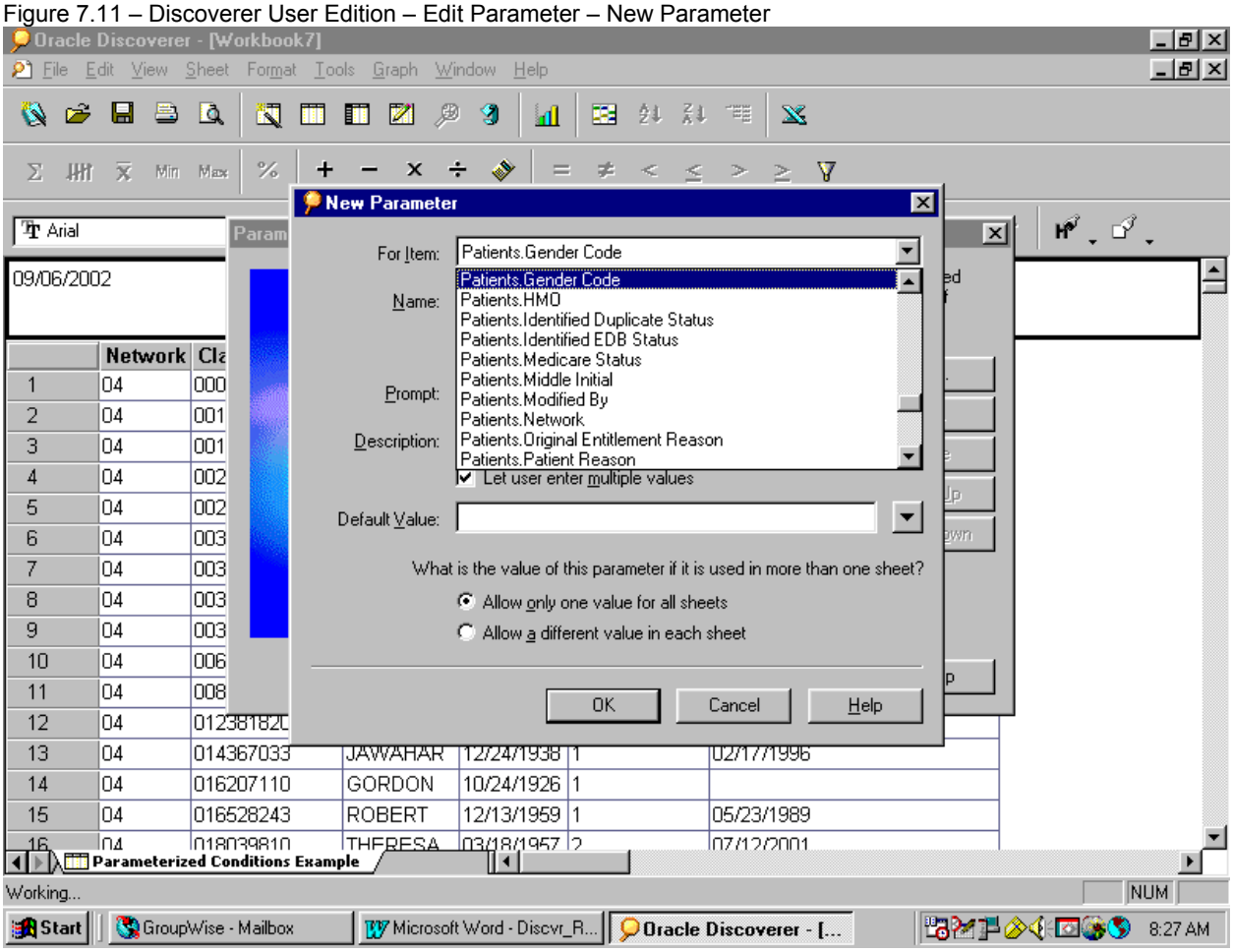
To create a new parameter from the Parameters tool, click on the New button to the right of the Parameters list box. The New Parameter dialog box displays. The window is nearly identical to the parameter entry box generated from the Conditions window, but there are a couple of important differences. First, the Item is now enabled for selection. Second, there is a new option to Create a condition with operator, and an entry field to select the operator. This basically does the reverse from our first parameters example. Instead of creating a parameter from the condition window, you can create a condition from the parameter window. The remaining items on the window - name, prompt, description, multiple values checkbox, and default value - are the same as what we have already covered. (Figure 7.10)

Figure 7.10 – Discoverer User Edition – Edit Parameter – New Parameter



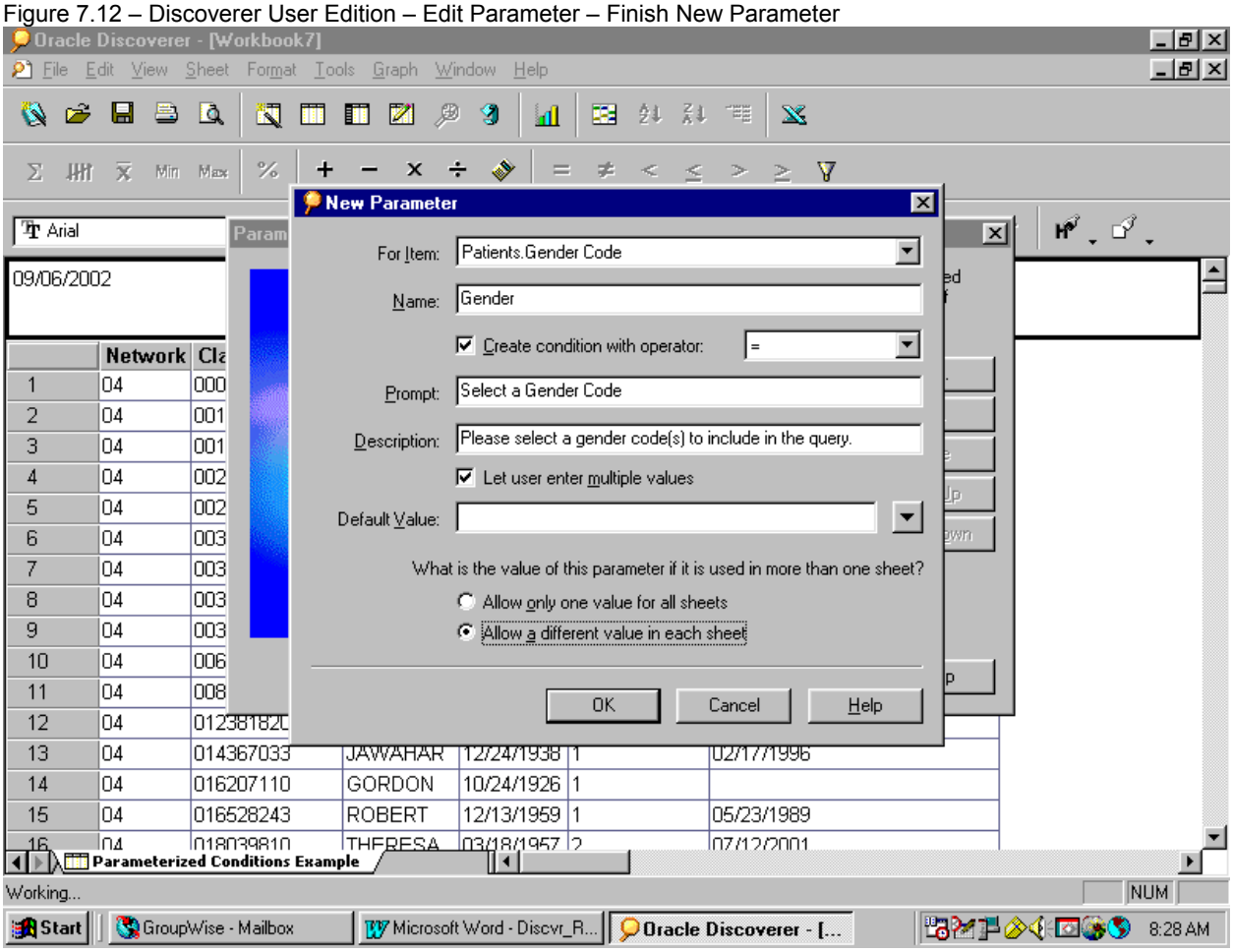
To associate the new parameter with an item from the folders in the current worksheet, select the item name from the Item dropdown list. (Figure 7.11)

Figure 7.11 – Discoverer User Edition – Edit Parameter – New Parameter



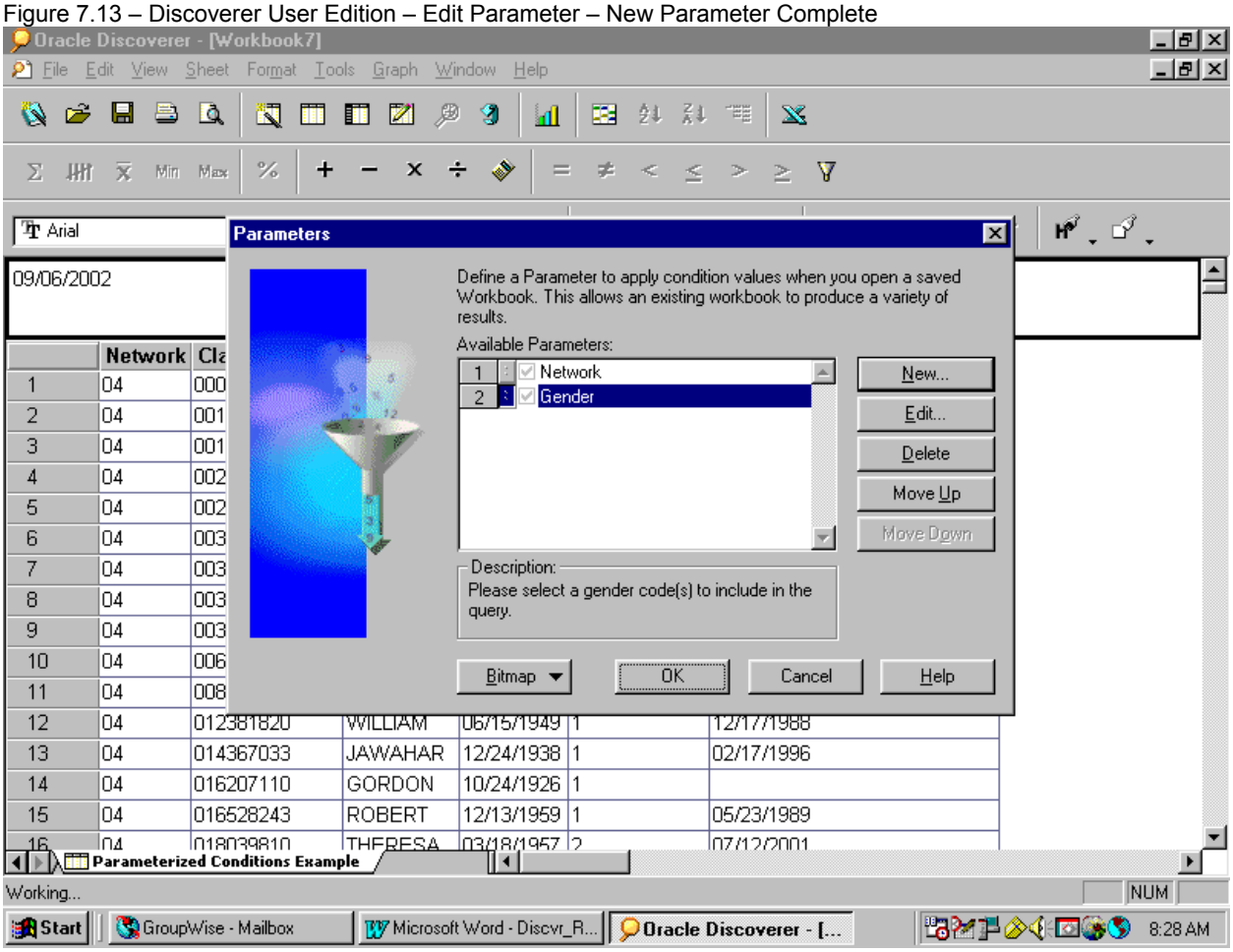
Enter the remaining information about the new parameter. To create a new condition automatically for the current worksheet using the new parameter, leave the Create condition option checked. If you are going to allow multiple values to be entered on the Parameter Wizard at run time, leave the Let user enter multiple values option checked. Click OK to return to the Parameters window. (Figure 7.12)

Figure 7.12 – Discoverer User Edition – Edit Parameter – Finish New Parameter



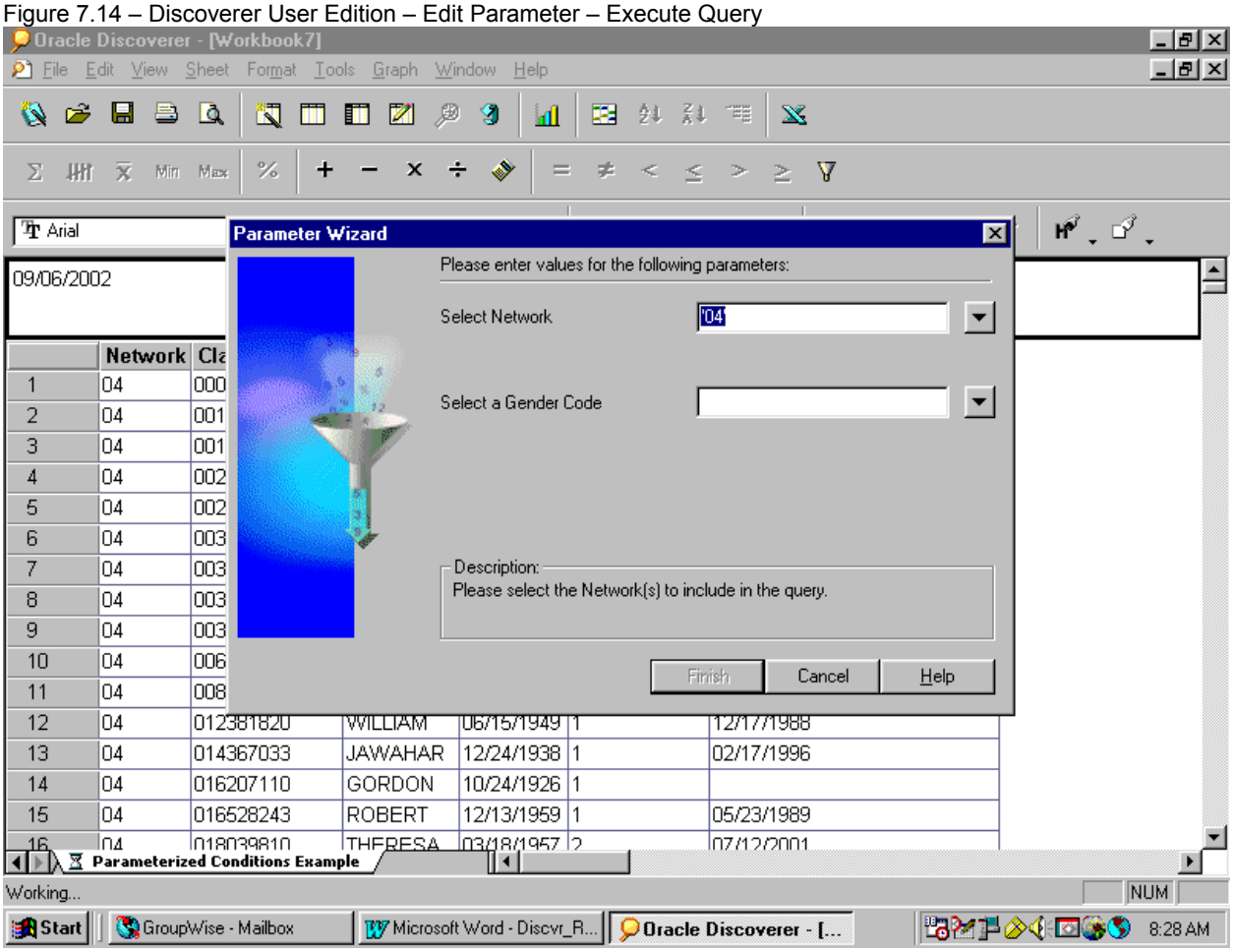
Upon returning to the Parameter window, you find the new parameter for entry of the Gender Code has been saved, a new condition created to use this parameter, and the Move Up button has now become enabled. To swap the order of the parameters for the sheet, highlight the parameter to be moved, then click the appropriate Move button. The order in which you see the parameters listed on this window is the order in which they are presented to the end user at runtime. Click OK to return to the worksheet. (Figure 7.13)

Figure 7.13 – Discoverer User Edition – Edit Parameter – New Parameter Complete



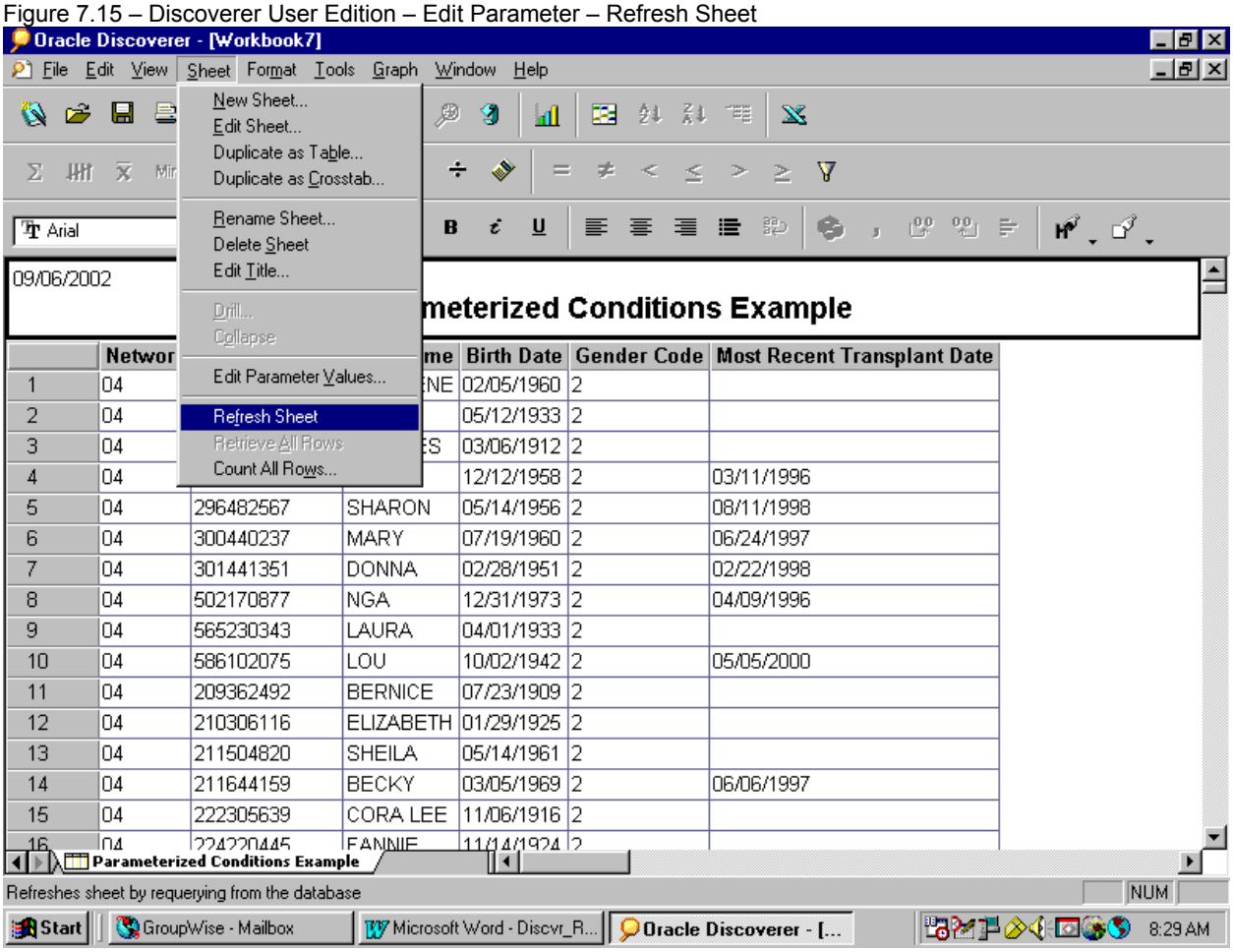
The Parameter Wizard appears again, this time with the extra parameter just coded. Now there are two required entries that must be selected prior to executing the query. The Finish button will not be enabled until all parameter items have been entered. Click Finish, and the query will run with the newly entered values. (Figures 7.14)

Figure 7.14 – Discoverer User Edition – Edit Parameter – Execute Query



To rerun the query at any time with new values in the parameters, click the Refresh Sheet icon in the Menu toolbar. The Parameter Wizard will display, and you may enter new values for the parameter(s) you wish to change. (Figures 7.15)

Figure 7.15 – Discoverer User Edition – Edit Parameter – Refresh Sheet



	Networ			me	Birth Date	Gender Code	Most Recent Transplant Date
1	04			NE	02/05/1960	2	
2	04				05/12/1933	2	
3	04			ES	03/06/1912	2	
4	04				12/12/1958	2	03/11/1996
5	04	296482567	SHARON		05/14/1956	2	08/11/1998
6	04	300440237	MARY		07/19/1960	2	06/24/1997
7	04	301441351	DONNA		02/28/1951	2	02/22/1998
8	04	502170877	NGA		12/31/1973	2	04/09/1996
9	04	565230343	LAURA		04/01/1933	2	
10	04	586102075	LOU		10/02/1942	2	05/05/2000
11	04	209362492	BERNICE		07/23/1909	2	
12	04	210306116	ELIZABETH		01/29/1925	2	
13	04	211504820	SHEILA		05/14/1961	2	
14	04	211644159	BECKY		03/05/1969	2	06/06/1997
15	04	222305639	CORA LEE		11/06/1916	2	
16	04	224220445	FANNIE		11/14/1924	2	

Lab Exercise 7 – Creating Parameters

For this lab exercise, we will add a parameter to a worksheet.

1. Create another copy of the sheet you completed in Lab 4, Simple Conditions Lab 4. When the Duplicate as Table window opens, make no changes to the sheet and click OK to close the window.
2. Rename the new sheet to Parameters Lab 7.
3. Open the Parameters tool by accessing the Tools menu item, then selecting Parameters from the dropdown list.
4. Click the New… button to create a new parameter.
5. Select item Patient.Birth Date from the Items dropdown list.
6. Name the parameter Starting Birth Date.
7. Leave the …create condition… box checked and change indicator to >=.
8. Enter “Select Starting Birth Date” into the Prompt field.
9. Enter “Please select starting birth date to include in the query.” into the Description field.
10. Click OK to save the parameter.
11. Add another new parameter. Click the New… button to create a new parameter.
12. Select item Patient.Birth Date from the Items dropdown list.
13. Name the parameter Ending Birth Date.
14. Leave the …create condition… box checked and change indicator to <=.
15. Enter “Select Ending Birth Date” into the Prompt field.
16. Enter “Please select ending birth date to include in the query.” into the Description field.
17. Click OK to save the parameter.
18. From the Parameters window, click OK to return to the worksheet.
19. Does the Parameter Wizard appear? How many parameters are listed?
20. Click Finish to return to the worksheet and run the query.
21. Click Refresh Sheet to rerun the query with different values for any of the parameters.